

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY  
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

**ÚSTAV TELEKOMUNIKACÍ**

DEPARTMENT OF TELECOMMUNICATIONS

**INTERAKTIVNÍ WEBOVÉ APLIKACE PRO PODPORU  
VÝUKY TEORIE BAREV**

INTERACTIVE WEB APPLICATIONS FOR EDUCATION OF COLOR THEORY

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Anfisa Koroleva**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Mgr. Pavel Rajmic, Ph.D.**

**BRNO 2021**

# Bakalářská práce

bakalářský studijní program **Telekomunikační a informační systémy**

Ústav telekomunikací

**Studentka:** Anfisa Koroleva

**ID:** 203494

**Ročník:** 3

**Akademický rok:** 2020/21

**NÁZEV TÉMATU:**

## Interaktivní webové aplikace pro podporu výuky teorie barev

### POKYNY PRO VYPRACOVÁNÍ:

Nastudujte problematiku vztahující se k online apletům týkajícím se výuky zpracování obrazů. Navrhněte funkcionalitu apletů jejich a ovládání, implementujte a odlaďte je. Půjde o interaktivní podání algoritmů používaných v oblasti počítačového zpracování barevných obrazů, konkrétně budou aplety zaměřeny na manipulace s barvami, barevné modely, jejich vztahy a převody: 1/ Míchání barev v různých modelech, 2/ Převod barevného obrazu na šedotónový, 3/ Převod šedotónového obrazu na barevný (tzv. nepravé barvy), 4/ Vnímání barvy člověkem (tj. převod fyzikálního světla do hodnot v modelu LAB, který modeluje lidské vnímání). Aplety implementujte pomocí HTML a JavaScriptu tak, aby demonstrovaly příslušnou teorii a vybízely studenty k interakci. Vytvořte kompletní HTML stránky včetně stručného shrnutí teorie.

### DOPORUČENÁ LITERATURA:

[1] Beneš, B.; Sochor, J.; Felkel, P.; Žára, J.: Moderní počítačová grafika. Computer Press, Brno, 2005.

[2] Gonzalez, R.C.; Woods, R.E.: Digital Image Processing. Třetí vydání. Pearson; 2007. ISBN 978-0131687288

**Termín zadání:** 1.2.2021

**Termín odevzdání:** 31.5.2021

**Vedoucí práce:** doc. Mgr. Pavel Rajmíc, Ph.D.

**prof. Ing. Jiří Mišurec, CSc.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## ABSTRAKT

Daná bakalářská práce se zabývá problematikou interpretace barev a barevných obrazů v počítačové grafice. Konkrétně je práce zaměřena na vývoj jednoduchých webových aplikací – appletů, které budou sloužit pro interaktivní podání metod používaných v oblasti počítačového zpracování barevných obrazů. Vytvořené applety by měly demonstrovat příslušnou teorii týkající se míchání barev v různých barevných modelech, převodu barevného obrazu na šedotónový, převodu šedotónového obrazu do nepravých barev, převodu fyzikálního světla do barevných prostorů CIE XYZ a  $L^*a^*b^*$ . V první části práce je popsán teoretický základ k požadovaným appletům. Následující praktická část je věnována vlastnímu postupu implementace appletů pomocí JavaScriptu a HTML.

## KLÍČOVÁ SLOVA

applet, barevná paleta, barevný model, barevný prostor, CIE XYZ, CIELAB, CMY, HSI, HTML, JavaScript, nepravá barva, pseudobarvení, RGB, standardní osvětlení, standardní pozorovatel, stupně šedi,  $xyY$ ,  $Y_C B_C R$

## ABSTRACT

This bachelor thesis deals with the issue of interpretation of colors and color images in computer graphics. Specifically, the work is focused on the development of simple web applications – applets that will serve for interactive submission of methods used in the field of computer processing of color images. The created applets should demonstrate the relevant theory relating to color mixing in different color models, conversion of a color image to a gray-scale image, conversion of a gray-scale image to a false-color image, conversion of physical light into CIE XYZ and  $L^*a^*b^*$  color spaces. The first part of the thesis describes the theoretical basis for the required applets. The following practical part is dedicated to the process of implementing the applets using JavaScript and HTML.

## KEYWORDS

applet, CIELAB, CIE XYZ, CMY, color model, color palette, color space, false color, gray scale, HSI, HTML, JavaScript, pseudocoloring, RGB, standard illuminant, standard observer,  $xyY$ ,  $Y_C B_C R$

KOROLEVA, Anfisa. *Interaktivní webové aplikace pro podporu výuky teorie barev*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2021, 51 s. Bakalářská práce. Vedoucí práce: doc. Mgr. Pavel Rajmic, Ph.D.

## Prohlášení autora o původnosti díla

**Jméno a příjmení autora:** Anfisa Koroleva  
**VUT ID autora:** 203494  
**Typ práce:** Bakalářská práce  
**Akademický rok:** 2020/21  
**Téma závěrečné práce:** Interaktivní webové aplikace pro podporu výuky teorie barev

Prohlašuji, že svou závěrečnou práci jsem vypracovala samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....  
.....  
podpis autorky\*

---

\*Autor podepisuje pouze v tištěné verzi.

## PODĚKOVÁNÍ

Ráda bych poděkovala vedoucímu bakalářské práce panu doc. Mgr. Pavlu Rajmicovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

# Obsah

<b>Úvod</b>	<b>10</b>
<b>1 Teoretický úvod</b>	<b>11</b>
1.1 Reprezentace obrazu v počítači . . . . .	11
1.2 Základní vlastnosti barevného obrazu v počítačové grafice . . . . .	12
1.2.1 Barva a její vlastnosti, barevný vjem člověka . . . . .	12
1.2.2 Barevné modely a prostory . . . . .	14
1.2.3 Převod barev mezi barevnými modely a prostory . . . . .	23
1.3 Šedotónový obraz v počítačové grafice . . . . .	28
1.4 Nepravé barvy . . . . .	29
1.4.1 Pseudobarvení . . . . .	29
<b>2 Praktická část</b>	<b>32</b>
2.1 Software . . . . .	32
2.1.1 HTML a CSS . . . . .	32
2.1.2 JavaScript . . . . .	32
2.1.3 Chart.js . . . . .	33
2.1.4 Editor zdrojového kódu VSCode . . . . .	33
2.2 Applet „Míchání barev v různých modelech“ . . . . .	34
2.3 Applet „Převod šedotónového obrazu na barevný (nepravé barvy)“ . . . . .	36
2.4 Applet „Převod barevného obrazu na šedotónový“ . . . . .	39
2.5 Applet „Převod fyzikálního světla do barevných prostorů CIE XYZ a $L^*a^*b^*$ “ . . . . .	42
<b>Závěr</b>	<b>46</b>
<b>Literatura</b>	<b>47</b>
<b>Seznam symbolů a zkratk</b>	<b>50</b>
<b>A Obsah přílohy</b>	<b>51</b>



# Seznam obrázků

1.1	Citlivost lidského oka na dopadající světlo. Převzato s úpravami z [3].	13
1.2	Geometrická reprezentace modelu RGB.	15
1.3	Geometrická reprezentace modelu CMY.	15
1.4	Reprezentace modelu HSI ve tvaru dvojitého kuželu.	17
1.5	Funkce standardního kolorimetrického pozorovatele CIE.	18
1.6	SPD funkce standardního osvětlovacího zdroje CIE D65.	19
1.7	Chromatický diagram CIE 1931. Původní obrázek převzat z [10].	21
1.8	Reprezentace modelu CIE $L^*a^*b^*$ pomocí barevné koule.	23
1.9	Vyhledávací tabulka LUT.	30
1.10	Ukázka převodu šedotónového obrazu do nepravých barev pomocí různých barevných palet: HSV, cool, hot, Jet.	31
2.1	Ukázka appletu „Míchání barev v různých modelech“.	34
2.2	Ukázka appletu pro převod šedotónových obrazů do nepravých barev.	36
2.3	Ukázka appletu pro převod barevného obrazu na šedotónový.	39
2.4	Graf funkce standardního osvětlení CIE A.	42
2.5	Graf funkce standardního pozorovatele CIE.	43
2.6	Spektrum odrazivosti perfektního difuzéru.	43
2.7	Grafické vyjádření tristimulu XYZ v appletu.	43
2.8	Princip odvození hodnot základního tristimulu XYZ v prostoru CIE XYZ.	44

# Seznam výpisů

2.1	Ukázka kódu funkce pro převod z RGB na $Y C_B C_R$ . . . . .	35
2.2	Ukázka kódu pro nastavení barevných složek modelu RGB. . . . .	35
2.3	Ukázka kódu pro výpočet výsledné RGB palety v závislosti na barvě koncových bodů. . . . .	36
2.4	Ukázka přiřazení barev dle barevné palety Hot. . . . .	37
2.5	Ukázka implementace gama korekce. . . . .	38
2.6	Ukázka kódu metody „Lumino“. . . . .	40
2.7	Ukázka kódu metody „Average“. . . . .	40
2.8	Ukázka kódu metody „% RGB“. . . . .	41
2.9	Ukázka kódu metody „Lightness“. . . . .	42

# Úvod

Hlavním cílem dané bakalářské práce je navrhnout a následně implementovat sadu webových aplikací, které mají za úkol ukázat základní principy práce s barvami v počítači a názorně demonstrovat metody zpracování barevných obrazů v počítačové grafice. Tyto aplikace mají být realizovány formou appletů v jazyce JavaScript a začleněny do klasické webové HTML stránky, což umožňuje jejich rychlé a snadné spouštění prostřednictvím internetového prohlížeče, který je v dnešní době součástí skoro každého počítače. Jednotlivé applety budou zaměřeny na manipulace s barvami, barevné modely, jejich vztahy a převody. První applet se použije pro vysvětlení mechanismu míchání barev napříč různými barevnými modely a prostory. Druhý poslouží jako ukázka principu převodu barevného obrazu na šedotónový. Třetí applet znázorní převod šedotónového obrazu do nepravých barev. A nakonec čtvrtý applet se zaměří na demonstraci převodu fyzikálního světla do hodnot v prostoru CIE XYZ a CIE  $L^*a^*b^*$ . Applety vytvořené v rámci zadaného tématu by pak měly posloužit jako interaktivní ukázka příslušné teorie a to zejména v rámci výuky předmětů zabývajících se problematikou zpracování barev a barevných obrazů v počítači.

Předkládaná bakalářská práce je rozdělená na teoretickou a praktickou část. V jednotlivých podkapitolách teoretické části se vysvětluje jak je interpretován obraz v počítači, definuje se pojem „barva“, její základní vlastnosti a vnímání barev člověkem. Dál jsou v textu popisované existující barevné modely a základní barevné parametry, na kterých jsou tyto modely postaveny. Taktéž jsou zde uvedené vztahy sloužící k převodu barev mezi jednotlivými modely. Poslední části teoretického úvodu se věnují šedotónovému obrazu a vysvětlení principu umělého barvení šedotónových obrazů pomocí nepravých barev.

Praktická část se zabývá rozбором implementace jednotlivých appletů, včetně ukázky jejich grafického rozhraní, popisu důležitých funkcí ze zdrojového kódu a jejich ovládání. Před tím se ještě uvádí stručný popis programovacích jazyků a vývojového prostředí, jež byly použité pro realizaci daných appletů.

# 1 Teoretický úvod

V této části se definují základní vlastnosti barevného obrazu v počítačové grafice, vysvětluje se, jak vzniká barva a jak je interpretována v různých barevných modelech a prostorech. Rovněž se zde mluví o šedotónovém obrazu a zásadách umělého barvení obrazu.

## 1.1 Reprezentace obrazu v počítači

Pojem „obraz“ nejde definovat jednoznačně, jelikož se jeho definice může lišit v závislosti na své aplikační oblasti. Nicméně pro formální vymezení za obraz budeme pokládat matematický model, který je představen spojitou skalární funkcí dvou proměnných – *obrazovou funkcí*  $f(x, y)$  (jde o statický obraz v ploše, např. na sítnici oka, na počítačové obrazovce apod.) [1].

V rámci počítačové grafiky pak mluvíme o obrazu digitálním<sup>1</sup>, jehož obrazová funkce je diskrétní a má zde podobu matice složené z konečného počtu obrazových prvků nebo elementů, z nichž každý má v obraze své určité umístění a číselnou hodnotu. Tyto obrazové elementy jsou známé jako *pixely* (z angl. picture element). Každý pixel obsahuje informaci o tzv. *barevné hloubce* – počet bitů použitých k popisu určité barvy v konkrétním bodě. V praxi počet bitů v pixelu – tím pádem i počet barev v obrazu – bývá různě omezen, např. vlastnostmi digitálního zařízení nebo aplikace, pomocí kterých se obraz vytváří či zpracovává. Záleží také na použitém barevném modelu, ve kterém je obraz uložen. V počítačové grafice se běžně setkáme např. s 24bitovou barevnou hloubkou, která se používá v jednom z nejrozšířenějších barevných modelů – RGB. Při této barevné hloubce lze v modelu RGB dosáhnout přibližně 16,7 milionu zobrazitelných barev (jsou to tzv. *pravé barvy*, angl. True Color). Často se také pracuje s 8bitovými obrazy ve *stupních šedi*, kdy počet barev (přesněji odstínů) je roven 256.

Ačkoli se obrazy v počítačové grafice vytváří pomocí matematických struktur a relací, lidská intuice a analýza obrazové informace stále hrají zásadní role při výběru vhodné techniky reprezentace obrazu v počítači, neboť se často tato volba provádí podle subjektivních vizuálních úsudků člověka [2]. Hlavní úlohou při převodu reálného obrazu, který má pro člověka psychofyzikální charakter, do digitálního tvaru je tedy stanovení relace mezi vstupními obrazovými daty a vytvořenými modely reálného světla napodobujícími lidské vnímání. Právě těmito modelům se budou věnovat následující podkapitoly dané bakalářské práce.

---

<sup>1</sup>V počítačové grafice se rozlišují dva typy digitálního obrazu: vektorový a bitmapový. Obvykle však pojmem „digitální obraz“ se myslí především obraz bitmapový, jenž je tvořen jednotlivými obrazovými body. V rámci textu dané bakalářské práce se také hovoří o bitmapovém obraze.

## 1.2 Základní vlastnosti barevného obrazu v počítačové grafice

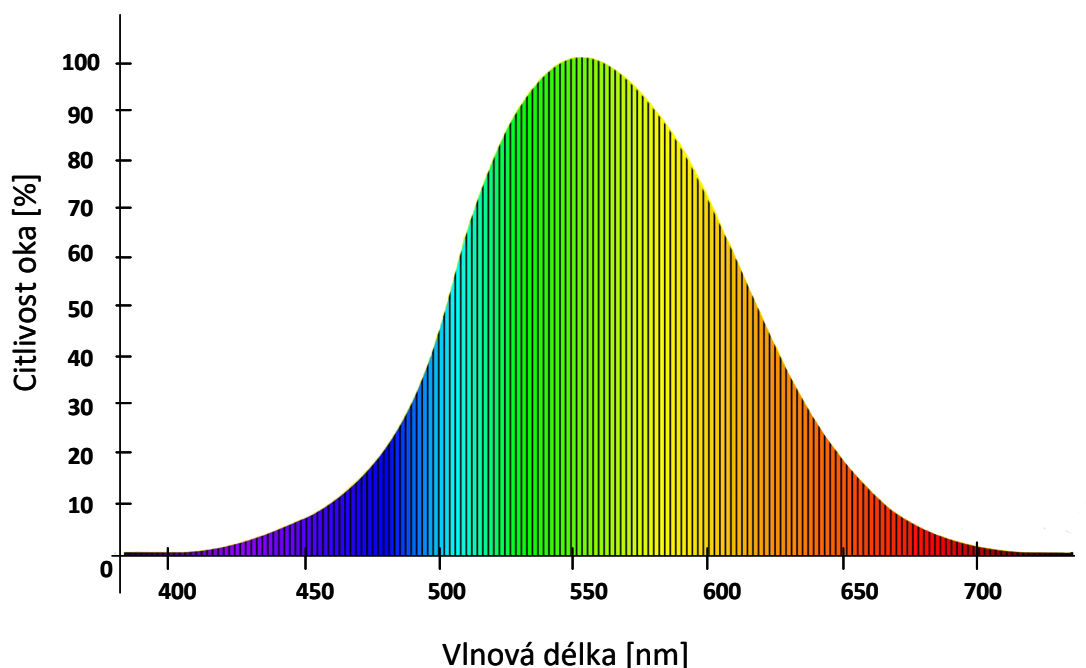
Než se dostaneme k barevným modelům a reprezentaci barevné informace v počítači, je potřeba vysvětlit co vlastně je barva, jak vzniká a jakým způsobem ji vnímá člověk.

### 1.2.1 Barva a její vlastnosti, barevný vjem člověka

Na pojem barva můžeme nahlížet z několika hledisek. Barvu lze chápat jako fyzikální vlastnost objektů, která je určena jejich schopností odrážet nebo pohlcovat viditelnou část elektromagnetického vlnění – světlo. Světlo dopadající na objekt se skládá z různých vlnových délek. Tím, že některé z nich se na povrchu objektu odrazí a některé se pohltí, vznikne barva objektu. Je také správné říct, že barva je vlastností světla samotného, jelikož přímo závisí na jeho spektrálním složení a zároveň i na jeho intenzitě. Nakonec důležitým faktorem při popisu barev je také subjektivní vnímání dané barvy pozorovatelem.

K barevnému vidění u člověka dochází prostřednictvím očních receptorů zvaných *čípky trojího druhu*. Tyto čípky se rozlišují podle relativní citlivosti na určité vlnové délky přicházejícího světla, přesněji pak na tři základní nebo tzv. *primární barvy* – červenou (R), zelenou (G) a modrou (B). Omezení citlivosti čípků do tří barev je ale docela formální záležitost, neboť nelze říci, že by čípek reagoval na snímání jen jedné konkrétní barvy. Ve skutečnosti tyto barevné receptory reagují prakticky na všechny vlnové délky ve viditelném spektru. Pouze platí, že je každý druh čípku pro vnímání určité vlnové délky světla nejcitlivější. Čípky nejsou v oku zastoupeny ve stejném poměru: přibližně 65% všech čípků je citlivých na červené světlo, 33% – na zelené světlo, a zhruba 2% jsou citlivá na modré [2]. Obrázek 1.1 ilustruje odezvu oka na dopadající světlo v závislosti na vlnové délce a zároveň demonstruje viditelné spektrum barev, které se pro člověka pohybuje v rozsahu 380–750 nm. Z daného obrázku je jednoznačně vidět, že člověk nejcitlivěji reaguje na zelené světlo a je tedy schopen v oblasti zelené rozeznat nejvíce odstínů. Lze tudíž odvodit, že celková citlivost oka na jednotlivé barvy nesouvisí přímo s poměrem zastoupení jednotlivých druhů čípků v oku. Ve skutečnosti má mechanismus barevného vidění člověka zcela odlišný a komplikovanější princip, než prostá reakce jednotlivých čípků na světlo.

Jakoukoliv barvu lze obvykle popsat parametry *jasu*, *odstínu* a *syty*. Jas vyjadřuje achromatickou („bezbarvou“) představu o intenzitě dopadajícího světla a je lidským okem vnímán jako samostatný parametr. V závislosti na velikosti intenzity se pozorované objekty, a tedy i jejich barva, jeví jasnější nebo tlumenější. Vztah mezi vnímaným jasnem a intenzitou světla je docela složitý, neboť fyzická veličina se lidským vizuálním systémem transformuje na subjektivní parametr, který prakticky



Obr. 1.1: Citlivost lidského oka na dopadající světlo. Převzato s úpravami z [3].

není možné měřit. Nicméně jas je jedním z klíčových faktorů při popisu barevného pocitu u člověka [2].

Odstín je určen dominantní vlnovou délkou dopadajícího světla. Podle odstínu se rozlišují a jmenují jednotlivé barvy – červená, žlutooranževá, růžová atd. Sytost se vztahuje k relativní čistotě barvy a udává množství příměsi šedi v poměru k odstínu. Dvojice těchto vlastností společně tvoří parametr zvaný *chromatičnost* neboli *barevnost*, a proto se barevnému světlu také často říká chromatické. Zmíněné parametry jasu a barevnosti slouží pro základní charakteristiku barvy a běžně se využívají zejména v barevných modelech, jež se snaží dosáhnout reprezentace barev, která by nejvíc odpovídala barevnému vidění člověka. Na parametrech jasu a barevnosti je např. založen model HSI. Principu oddělení jasu jako samostatného parametru se využívá také v barevných prostorech CIE  $xyY$  a CIE  $L^*a^*b^*$ , v modelu  $YCbCr$ . Podrobněji se o tom ještě mluví v následující části o barevných modelech a prostorech.

## 1.2.2 Barevné modely a prostory

K reprezentaci barevného obrazu a jeho vlastností v počítači se používají *barevné modely* (též barevný systém) – matematické struktury, ve kterých se barvy vyjadřují kombinací určitých barevných složek – obvykle mísením základních barev nebo podílem základních barevných parametrů. Většina barevných modelů k reprezentaci jednotlivých barev používá nejméně tři barevné složky. Každá z nich je v rámci barevného modelu dána číselnou hodnotou. O hodnotách barevných složek lze také přemýšlet jako o souřadnicích barvy. Barevný model pak lze popsat jako souřadnicový systém, v rámci kterého je každá konkrétní barva reprezentována jediným bodem, jenž má své určité souřadnice [2].

Na barevném modelu často bývá založen *barevný prostor* – definována daným barevným modelem množina barev. Barevný prostor detailně popisuje reprezentaci jednotlivých barevných složek, navíc také standardizuje odstíny základních barev modelu [4].

Každý barevný model je postaven na různých parametrech a slouží k rozličným účelům. V dnešní době existuje celá řada barevných modelů a prostorů, většina z nich je však buď navržena pro určitá zařízení (barevné monitory, tiskárny atd.) nebo je zaměřena na aplikace pracující s barevnou grafikou (software pro vytváření animace, editory obrazu apod.) [2]. Nezávisle na oblasti, ve které je barevný model či prostor použit, jeho hlavním úkolem je především stanovit shodu mezi reálnými barvami vnímanými lidmi a barvami tvořenými na výstupních zařízeních.

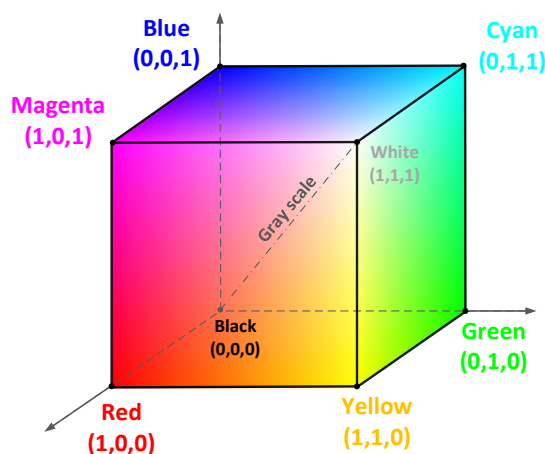
### Model RGB

Název modelu **RGB** pochází z počátečních písmen jeho jednotlivých barevných složek: **R** – red (červená), **G** – green (zelená) a **B** – blue (modrá). Samotný model je pak tvořen aditivním<sup>2</sup> mícháním těchto primárních barev a používá se ve většině monitorů (i ostatních zobrazovacích zařízeních) a také v digitálních kamerách, skenerech atd. pro reprezentaci barevného obrazu. Daný barevný model patří k modelům závislým na konkrétním zařízení, jelikož různá zařízení mají odlišné technické charakteristiky a jejich rozsahy zobrazitelných barev se mohou významně lišit.

Model RGB obvykle bývá reprezentován jednotkovou krychlí (viz obr. 1.2), ve které jednotlivé složky R, G a B nabývají hodnot v rozsahu  $[0, 1]$ , neboli 0–255 v celočíselném rozsahu. Počátek souřadnic krychle  $(0, 0, 0)$  odpovídá černé barvě a bod vrcholu o souřadnicích  $(1, 1, 1)$  tvoří bílou barvu. Na obrázku 1.2 si také můžeme všimnout barvy sekundární: Cyan (azurová), Magenta (purpurová) a Yellow (žlutá).

---

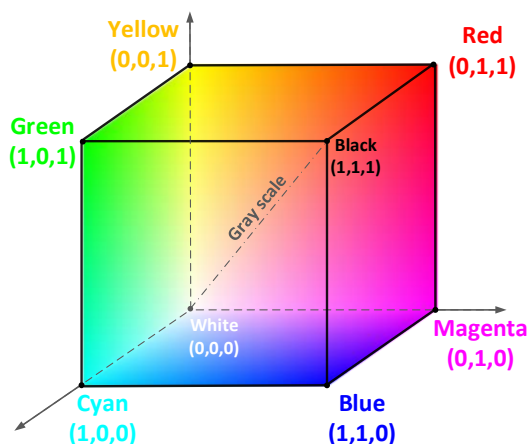
<sup>2</sup>Aditivní míchání – způsob míchání barev, kdy se jednotlivé složky barev sčítají. Jde o míchání vyzařovaného světla. Smícháním tří základních barev vzniká bílá barva.



Obr. 1.2: Geometrická reprezentace modelu RGB.

### Model CMY(K)

Daný barevný model je založen na subtraktivním<sup>3</sup> míchání barev. **CMY** má opačný vztah k modelu RGB: hlavními barvy jsou tedy doplňkové barvy modelu RGB, tedy azurová (C – cyan), purpurová (M – magenta) a žlutá (Y – yellow). Patří rovněž k barevným modelům závislým na zařízení. Obdobně jako RGB lze model CMY umístit do jednotkové krychle (obr. 1.3), opět za předpokladu, že hodnoty základních barevných složek jsou normalizovány v intervalu  $[0, 1]$ .



Obr. 1.3: Geometrická reprezentace modelu CMY.

<sup>3</sup>Subtraktivní míchání – mícháním barvy od sebe odčítáme (omezujeme barevné spektrum, které se odráží od povrchu). Vzájemným smícháním základních barev teoreticky vzniká černá barva.



V praxi se často můžeme setkat s modelem CMYK, který vzniká přidáním černé (K – black) jako čtvrté základní barvy. Daný barevný model je navržen pro práci s tiskovými zařízeními, jelikož při použití reálných barviv subtraktivním mícháním barev CMY nejde dosáhnout čisté černé barvy. Azurová, purpurová a žlutá ve skutečnosti dokáží vytvořit jenom špinavě hnědou barvu. Zároveň je využití samostatné černé barvy oproti míchání všech jednotlivých barev výrazně ekonomičtější. Navíc lze její pomocí ztmavit i ostatní barvy [1].

Modely RGB a CMY jsou vhodné především pro generování barev – jako např. při zachycení obrazu digitální kamerou nebo při zobrazení obrazu na monitoru. Avšak jejich použití pro popis barev je méně užitečné. Reprezentace barev ve tvaru jednotkové krychle je výhodou, pokud jde o zpracovávání barev pouze počítačem. Ke komplikacím dochází v případě, když má zadat do počítače barvu v prostoru RGB či CMY přímo uživatel, neboť představy o tom, jaká barva vznikne kombinací základních složek, jsou pro každého člověka poměrně odlišné [1]. Proto za účelem více přizpůsobit popis barev lidskému vnímání vznikly další barevné modely a prostory.

## Model HSI

Jedním z modelů sloužících člověku k popisu barev intuitivnějším způsobem je barevný model **HSI**, který odděluje údaje o jasů od informace o jejích chromatických vlastnostech. Technicky je to model závislý na zařízení, nicméně je velmi užitečný pro úpravy RGB obrázků. Obvykle je používán pro nastavení barvy v různých grafických programech.

Barevný model HSI je založen na třech hlavních barevných parametrech:

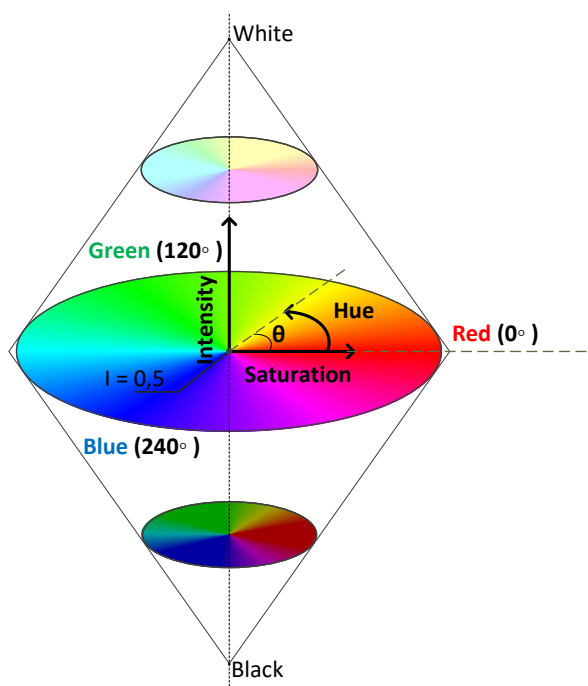
- **Hue (H)** – *odstín*. Reprezentuje převládající spektrální barvu v rozsahu od 0 do 360 stupňů. Obvykle bývá zobrazován jako kruh, ve kterém každá základní barva má svůj daný úhel: 0° se rovná červené barvě, 120° je zelená, 240° odpovídá modré barvě, 60° je žlutá, 300° – purpurová [5];
- **Saturation (S)** – *sytylost*. Vyjadřuje „čistotu“ barvy a udává poměr příměsi odstínů šedé (včetně černé a bílé) v čisté barvě.

Komponenta S nabývá hodnot v rozsahu [0, 1], kde 0 představuje černou, 1 – bílou barvu. Sytylost lze měřit i v procentech: 100% sytylost – bílá barva, 0% sytylost odpovídá černé barvě;

- **Intensity (I)** – *intenzita*. Vyjadřuje postavení konkrétní barvy vzhledem ke stupnici šedé – od světlé po tmavou. Jinými slovy, označuje stupeň tmavosti či světlosti barevného odstínu. Intenzitu lze jednoduše interpretovat číselnými hodnotami v rozsahu 0 (zastupuje černou barvu) až 1 (zastupuje bílou), nebo stejně jako pro sytylost je možné při vyjádření intenzity použít procenta.

Obr. 1.4 znázorňuje model HSI v podobě dvojitého kuželu, ze kterého lze vyčíst

jednotlivé parametry pro zvolenou barvu, tedy její odstín, sytost a intenzitu. Barevný odstín je vyjádřen úhlovou hodnotou  $\theta$ , intenzita se mění od 0 (černá v dolním vrcholu) do 1 (bílá v horním vrcholu). Sytost nabývá na povrchu kuželů hodnoty 1 a klesá na 0 směrem k centrální ose kuželů. Nejjasnější čisté barvy mají tedy souřadnice  $S = 1$  a  $I = 0,5$  a leží na obvodu podstavy kuželů.



Obr. 1.4: Reprezentace modelu HSI ve tvaru dvojitého kuželu.

### Model $YC_B C_R$

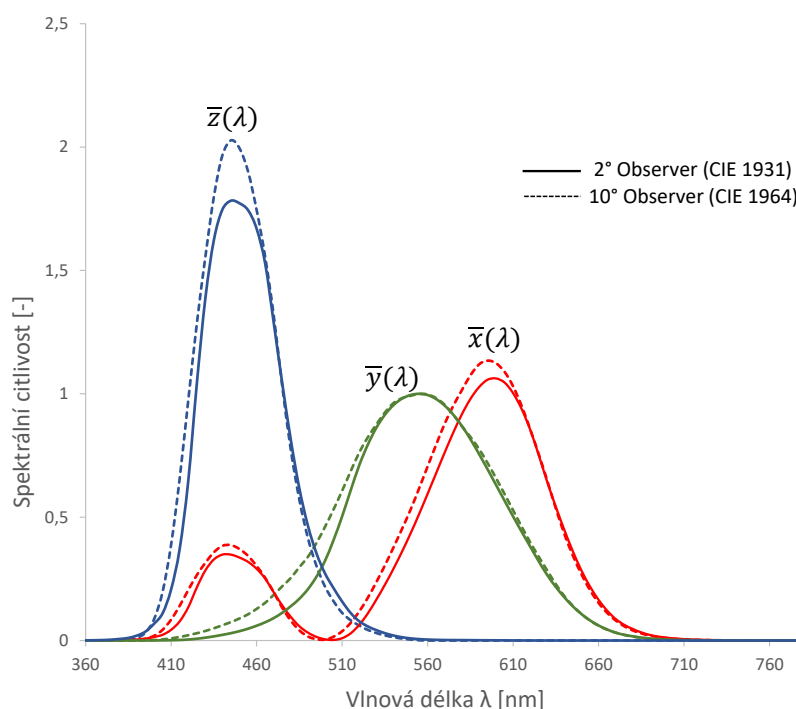
Daný model je jedním z barevných modelů používaných primárně k reprezentaci a zpracování digitálního videa či obrazu. V počítačové grafice se s  $YC_B C_R$  můžeme setkat při zápisu bitmapových obrázků ve formátu JPEG. Model se také používá při MPEG kompresi formátů Blu-ray, DVD, Video CD a při kódování audiovizuálních signálů v oblasti digitální televize [6]. Základem barevného modelu  $YC_B C_R$  je oddělení jasové (luminanční) složky označené  $Y$  od informace o barvě (chrominance), která je reprezentována dvěma chroma parametry  $C_B$  a  $C_R$ , přičemž  $C_B$  je rozdíl modré a lumy ( $B - Y$ ),  $C_R$  – rozdíl červené a lumy ( $R - Y$ ).

## Prostor CIE XYZ

Barevný prostor **CIE XYZ** (též CIE 1931) byl vyvinut Mezinárodní komisí pro osvětlování (CIE) v roce 1931 na základě studií o barevném vidění „průměrného“ člověka. Hlavním cílem organizace CIE bylo vytvořit „univerzální barevný systém“, který by systematicky charakterizoval světelné a pozorovací podmínky pro sledování a měření barev a stanovil metodiku k odvození hodnot pro popis barvy bez ohledu na oblast nebo případ použití [7].

S tímto účelem byly komisí CIE v rámci barevného systému CIE XYZ specifikovány *standardní osvětlovací zdroj* (angl. standard illuminant) a *standardní pozorovatel* (angl. standard observer).

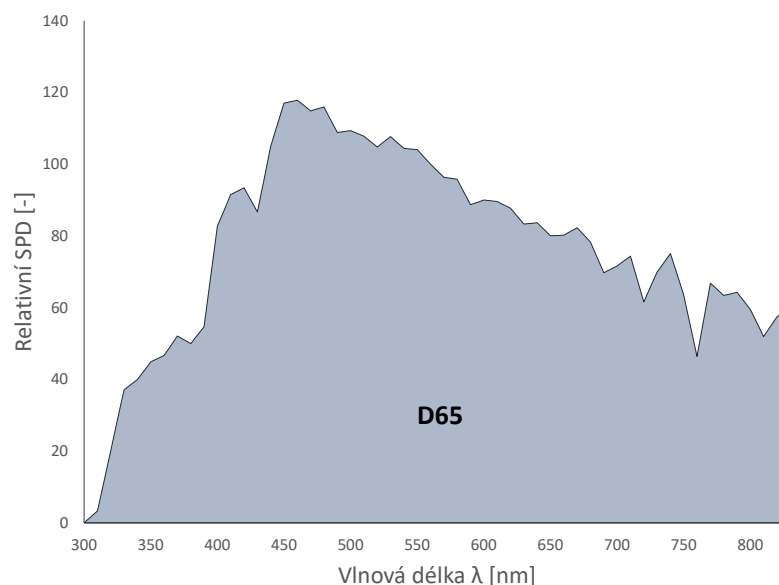
- **Standardní pozorovatel** definuje hypotetického „průměrného“ lidského pozorovatele. Matematickým vyjádřením standardního pozorovatele jsou tzv. *barvové funkce* (angl. color-matching functions), které reprezentují odezvu „normálního“ lidského vizuálního systému na různé vlnové délky barevného světla ve viditelném spektru. V roce 1931 byl definován standardní pozorovatel s 2° zorným polem a v roce 1964 byla přidána definice doplňkového standardního pozorovatele s 10° zorným polem. Tyto standardy se běžně používají i dodnes.



Obr. 1.5: Funkce standardního kolorimetrického pozorovatele CIE.

- **Standardní osvětlovací zdroj** - specifikace teoretických světelných zdrojů, které se používají pro porovnávání barev. Komisí CIE jsou definována standardní osvětlení A, B a C, hypotetické osvětlení E a sady osvětlení D a F. Standardní osvětlovací zdroje A–C jsou charakterizované jako žárovky, které reprodukuje světlo určité barevné teploty<sup>4</sup>. Skupina F (F1–F12) představuje zářivkové osvětlení, jež se také rozlišují podle barevné teploty vyzařovaného světla. Osvětlení E definuje teoretický světelný zdroj s rovnoměrným rozložením energie. Ve skutečnosti však toto osvětlení neexistuje. Skupina standardních osvětlení D reprezentuje denní světlo a zahrnuje světelné zdroje D50, D65 a D75 (číslo za písmenem je zkratkou barevné teploty zdroje: 5000 K, resp. 6500 K a 7500 K). Mezi těmito standardními osvětlovacími zdroji D65 je nejvíce přiblížen k reálnému nepřímému slunečnímu světlu a spolu se standardním osvětlením D50 je dnes při posuzování barev zdaleka nejpreferovanější a nejrozšířenější [8].

Grafickým vyjádřením standardního osvětlovacího zdroje je SPD funkce – spektrální rozložení energie světla (angl. spectral power distribution), které vyzařuje daný zdroj, v závislosti na vlnové délce.



Obr. 1.6: SPD funkce standardního osvětlovacího zdroje CIE D65.

<sup>4</sup>Zdroje vyzařující světlo mohou být popsány pomocí barevné teploty podobně jako všechna zahřátá tělesa, která také vyzařují světlo [8]. V závislosti na velikosti barevné teploty se světlo zdá „teplejší“ nebo „studenější“.

Na základě barvových funkcí standardního pozorovatele a SPD funkce standardního osvětlení se odvodí číselné hodnoty tristimulu XYZ, které v prostoru CIE XYZ slouží pro reprodukci všech existujících barev viditelného spektra. Většinou hodnoty XYZ nachází uplatnění v kolorimetrii při měření barev různých objektů. Barvy zadané trojicí XYZ v prostoru CIE XYZ nejsou vázány na žádné zařízení. Jednotlivé složky X, Y a Z jsou definovány následujícím způsobem [9]:

$$X = 100 \cdot \frac{\int_{\lambda} E(\lambda) R(\lambda) \bar{x}(\lambda) d\lambda}{\int_{\lambda} E(\lambda) \bar{y}(\lambda) d\lambda}, \quad (1.1a)$$

$$Y = 100 \cdot \frac{\int_{\lambda} E(\lambda) R(\lambda) \bar{y}(\lambda) d\lambda}{\int_{\lambda} E(\lambda) \bar{y}(\lambda) d\lambda}, \quad (1.1b)$$

$$Z = 100 \cdot \frac{\int_{\lambda} E(\lambda) R(\lambda) \bar{z}(\lambda) d\lambda}{\int_{\lambda} E(\lambda) \bar{y}(\lambda) d\lambda}, \quad (1.1c)$$

kde  $E(\lambda)$  – SPD funkce standardního osvětlení;

$R(\lambda)$  – spektrální odrazivost měřeného objektu při vlnové délce  $\lambda$ ;

$\bar{x}(\lambda)$ ,  $\bar{y}(\lambda)$ ,  $\bar{z}(\lambda)$  – barvové funkce standardního pozorovatele.

Přestože je tristimulus XYZ přímou číselnou reprezentací změřené barvy objektu, není moc pro člověka vhodný k intuitivnímu popisu barev. Proto se v praxi hodnoty tristimulu XYZ zpravidla používají jako referenční hodnoty pro transformace barev a slouží jako základ pro další barevné prostory.

## Prostor xyY

V trojici XYZ hodnota Y zastupuje jas. Tento parametr lze zafixovat a přesunout se do prostoru **xyY**. Daný barvový prostor je navržen tak, že parametr Y zde separuje jasovou složku a samotná barva je pak definována tzv. *souřadnicemi chromatičnosti*  $x$ ,  $y$  a  $z$ , které se vypočítávají ze složek XYZ následovně:

$$x = \frac{X}{X + Y + Z}, \quad (1.2a)$$

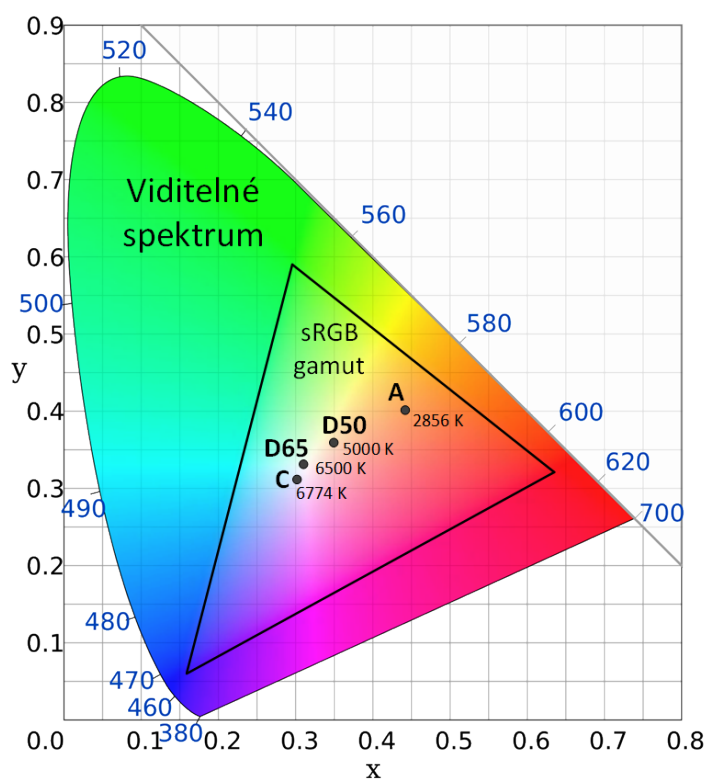
$$y = \frac{Y}{X + Y + Z}, \quad (1.2b)$$

$$z = \frac{Z}{X + Y + Z}. \quad (1.2c)$$

K úplnému určení barvy v prostoru xyY však stačí pouze dvě souřadnice –  $x$  a  $y$  – doplněné o jasovou hodnotu Y. Z rovnic (1.2a)–(1.2c) plyne, že  $x + y + z = 1$ , a proto

hodnotu souřadnice  $z$  lze dle potřeby jednoduše odvodit ze zbývajících dvou jako  $z = 1 - x - y$ .

Hodnoty prostoru  $xyY$  slouží především pro grafické vyjádření barev ve dvojrozměrném prostoru, kterému se říká *chromatický diagram* (viz obr. 1.7). Barvy jsou v tomto diagramu reprezentované bez ohledu na jas, tedy pouze pomocí chromatických souřadnic  $x$  a  $y$  (jasová složka  $Y$  pak slouží k odvození původních hodnot  $XYZ$ ). Sytost barvy se mění od středu diagramu směrem k jeho obvodu (na okrajích se pak nacházejí nejsytější barvy). Při pohybu po obvodu diagramu se mění barevný odstín. Chromatický diagram názorně ilustruje všechny viditelné barvy a jejich vzájemné vztahy. Rovněž lze v diagramu označit barevné teploty standardních osvětlení, jež ukazují jaký barevný odstín bude mít světlo reprodukováné danými osvětlovacími zdroji. Navíc pomocí chromatického diagramu můžeme jednoduše zobrazit tzv. *barevný gamut* – rozsah barev, který je schopen pojmout určitý barevný prostor. Na obrázku 1.7 je jako příklad uveden gamut jednoho v praxi z nejpoužívanějších barevných prostorů – sRGB. Gamut lze stanovit také pro všechna zařízení pracující s barevnými prostory, v nichž jsou barvy definovány pomocí hodnot jakéhosi tristimulu [8]. Barevný gamut pomáhá vymezit oblast barev reprodukovatelných určitým zařízením a běžně se používá pro potřeby počítačového zpracování, zobrazování či tisku obrazu.



Obr. 1.7: Chromatický diagram CIE 1931. Původní obrázek převzat z [10].

## Prostor CIE L\*a\*b\*

Prostor **CIELAB** (též L\*a\*b\*) je dalším barevným prostorem odvozeným z CIE XYZ. Barva je v prostoru L\*a\*b\* definována pomocí následujících souřadnic:

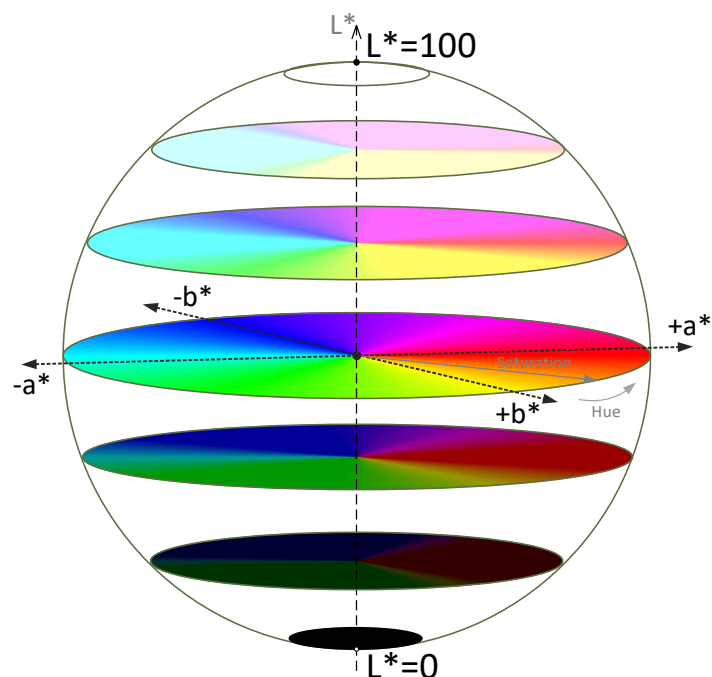
- **L\*** (Luminance) – světelnost – popisuje jas barvy v konkrétním bodě, nabývá hodnot v rozsahu 0–100, kde 0 = černá, 100 = bílá;
- barvosné složky **a\*** a **b\*** – popisují barvu v konkrétním bodě, přičemž složka **a\*** zastupuje barvy v rozmezí zelené až červené (+a = víc do červené, -a = víc do zelené), **b\*** – v rozmezí modré až žluté (+b = víc do žluté, -b = víc do modré). Souřadnice **a\*** může nabývat hodnot od -128 (zelená) do 128 (červená). Souřadnice **b\*** také nabývá hodnot od -128 (modrá) do 128 (žlutá).

Prostor L\*a\*b\* je navržen jako perceptuálně uniformní barvový prostor. To znamená, že číselná kolorimetrická vzdálenost mezi jednotlivými barvami zde odpovídá (alespoň do určité míry) pozorovatelem vnímané „blízkosti“ nebo „vzdálenosti“ těchto barev. Jednou z klíčových vlastností prostoru CIELAB je, že na bázi jeho základních parametrů lze vypočítat tzv. *barevný rozdíl*  $\Delta E$  (angl. color difference), který stanoví vzájemnou „vzdálenost“ dvou barev při jejich porovnání v prostoru L\*a\*b\*. Hodnota  $\Delta E$  je užitečná např. v případě, když je potřeba přesně vyjádřit barevnou toleranci nějakého zařízení [8].

$$\Delta E = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2}, \quad (1.3)$$

kde  $\Delta L^*$ ,  $\Delta a^*$ ,  $\Delta b^*$  – rozdíly souřadnic  $L^*$ ,  $a^*$  a  $b^*$  porovnávaných barev.

Další důležitou vlastností daného barevného prostoru je možnost individuálně nastavovat jeho jednotlivé složky. Touto vlastností nám prostor L\*a\*b\* umožňuje např. změnit barevný tón obrazu, aniž by současně došlo ke změně jeho jasu. Výhody praktického využití daného barevného prostoru se zejména projevují v práci s nástroji pro úpravu obrazů nebo editory fotografií, například při doostřování fotografií ve Photoshopu. Pro tento účel se RGB snímek převede do prostoru L\*a\*b\* a poté se doostří pouze parametr  $L$ , tj. pouze jasový kanál a nikoliv barvy, což ve výsledku redukuje vznik nežádoucích barevných artefaktů na hranách [11].



Obr. 1.8: Reprezentace modelu CIE  $L^*a^*b^*$  pomocí barevné koule.

### 1.2.3 Převod barev mezi barevnými modely a prostory

V této části se uvádí vzorce pro přepočítání jednotlivých barevných složek mezi popsanými barevnými modely a prostory, přesněji pak vzorce pro převod z modelu RGB do ostatních barevných modelů a prostorů. Uvedené matematické vztahy slouží jako základ pro applety vytvořené v rámci této bakalářské práce, zejména se využívají v appletu „Míchání barev v různých modelech“.

#### Převod barev mezi RGB a CMY

Převod mezi modely RGB a CMY je dán jednoduchým odčítáním tříprvkových matic o složkách CMY a RGB nabývajících hodnot v rozsahu 0 až 1:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \quad (1.4)$$

Z rovnice (1.4) snadno získáme opačný vztah pro převod barev z modelu CMY do modelu RGB.



## Převod barev mezi RGB a HSI

Převod barvy z modelu RGB do modelu HSI je definován jako [2]:

$$\begin{cases} H = \begin{cases} \theta, & \text{pokud } B \leq G \\ 360^\circ - \theta, & \text{pokud } B > G \end{cases} \\ \text{kde } \theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{\frac{1}{2}}} \right\} \\ S = 1 - \frac{3}{R + G + B} \min(R, G, B) \\ I = \frac{R + G + B}{3} \end{cases} . \quad (1.5)$$

V případě převodu hodnot z barevného modelu HSI do modelu RGB se berou v úvahu tři jednotlivé barevné úseky, kterým odpovídá určitý rozsah hodnot barevného odstínu H [2].

Pro barvy v rozmezí RG platí ( $0^\circ \leq H < 120^\circ$ ):

$$\begin{cases} B = I(1 - S) \\ R = I \left[ 1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\ G = 3I - (R + B) \end{cases} . \quad (1.6)$$

V úseku GB ( $120^\circ \leq H < 240^\circ$ ) se nejdříve musí přepočítat hodnota parametru H:

$$H = H - 120^\circ, \quad (1.7)$$

RGB komponenty se vypočtou následovně:

$$\begin{cases} R = I(1 - S) \\ G = I \left[ 1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\ B = 3I - (R + G) \end{cases} . \quad (1.8)$$

Nakonec v rozmezí BR ( $240^\circ \leq H < 360^\circ$ ) se hodnoty RGB vypočítají následovně (zde opět musíme přepočítat hodnotu H):

$$H = H - 240^\circ, \quad (1.9)$$

$$\begin{cases} G = I(1 - S)B \\ B = I \left[ 1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\ R = 3I - (G + B) \end{cases} \quad (1.10)$$

Všechny uvedené vztahy platí za předpokladu, že hodnoty složek RGB jsou normalizovány v rozsahu  $[0,1]$  a úhel  $\theta$  se měří vůči ose červené v prostoru HSI (jak je znázorněno na obr. 1.4). Parametry  $S$  a  $I$  pak nabývají hodnot  $0-1$  (nebo  $0$  až  $100\%$ ), odstín  $H$  má hodnoty  $0^\circ-360^\circ$  nebo lze  $H$  také normalizovat na rozsah  $[0, 1]$  vydělením  $360^\circ$ .

### Převod do modelu $YC_B C_R$

Jednou z možností převodu do  $YC_B C_R$  je vztah (1.11) definovaný standardem ITU-R BT.601 pro 8bitové vzorky barev:

$$\begin{cases} Y' = 16 + (65,481 \cdot R' + 128,553 \cdot G' + 24,966 \cdot B') \\ C_B = 128 + (-37,797 \cdot R' - 74,203 \cdot G' + 112,0 \cdot B') \\ C_R = 128 + (112,0 \cdot R' - 93,786 \cdot G' - 18,214 \cdot B') \end{cases} \quad (1.11)$$

Existují i další standardy, jež stanovují jiné poměry složek RGB, ve vztahu (1.11) je však uvedeno jedno z nejuniverzálnějších a běžně používaných rozvržení.

Pro opačný převod platí:

$$\begin{cases} R' = 298,082 \cdot Y' + 408,583 \cdot C_R - 222,921 \\ G' = 298,082 \cdot Y' - 100,291 \cdot C_B - 208,120 \cdot C_R + 135,576 \\ B' = 298,082 \cdot Y' + 516,412 \cdot C_B - 276,836 \end{cases} \quad (1.12)$$

### Převod barev mezi RGB a $L^*a^*b^*$

Převod z modelu RGB do prostoru CIELAB není možné provést přímo, proto nejprve za pomoci transformační matice  $\mathbf{M}$ , jež definována v (1.14), musíme hodnoty složek RGB převést na příslušné hodnoty prostoru CIE XYZ a pouze tehdy se dostaneme do CIELAB. Pro možnost použití převodní matice hodnoty složek RGB musí být normalizovány v rozsahu  $[0,1]$  [12]. Tím pádem:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = [\mathbf{M}] \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1.13)$$

$$[M] = \begin{bmatrix} S_r X_r & S_g X_g & S_b X_b \\ S_r Y_r & S_g Y_g & S_b Y_b \\ S_r Z_r & S_g Z_g & S_b Z_b \end{bmatrix}, \quad (1.14)$$

kde

$$\begin{cases} X_r = x_r/y_r; & Y_r = 1; & Z_r = (1 - x_r - y_r)/y_r \\ X_g = x_g/y_g; & Y_g = 1; & Z_g = (1 - x_g - y_g)/y_g \\ X_b = x_b/y_b; & Y_b = 1; & Z_b = (1 - x_b - y_b)/y_b \end{cases}. \quad (1.15)$$

$$\begin{bmatrix} S_r \\ S_g \\ S_b \end{bmatrix} = \begin{bmatrix} x_r & x_g & x_b \\ y_r & y_g & y_b \\ z_r & z_g & z_b \end{bmatrix}^{-1} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix}. \quad (1.16)$$

Zjištěné hodnoty XYZ převedeme na jednotlivé složky barevného prostoru  $L^*a^*b^*$  následujícím způsobem [2]:

$$\begin{cases} L^* = 116 \cdot h\left(\frac{Y}{Y_W}\right) - 16 \\ a^* = 500 \left[ h\left(\frac{X}{X_W}\right) - h\left(\frac{Y}{Y_W}\right) \right] \\ b^* = 200 \left[ h\left(\frac{Y}{Y_W}\right) - h\left(\frac{Z}{Z_W}\right) \right], \end{cases}. \quad (1.17)$$

$$\text{kde } h(q) = \begin{cases} \sqrt[3]{q} & q > 0,008856 \\ 7,787q + 16/116 & q \leq 0,008856 \end{cases}$$

Mezilehlé parametry:

- $S_r, S_g, S_b$  – součet hodnot tristimulu XYZ základních barevných složek;
- $(x_r \ y_r), (x_g \ y_g), (x_b \ y_b)$  – souřadnice chromatičnosti základních barevných složek;
- $X_W Y_W Z_W$  – tzv. *bod bílý* neboli *referenční bílá* (angl. white point, resp. reference white) – hodnoty tristimulu CIE XYZ pro ideální bílé těleso dokonale odrážející dopadající světlo. Tyto hodnoty se liší v závislosti na použitém standardním osvětlení.

Opačná transformace z prostoru CIELAB do modelu RGB by pak zase provedla pomocí hodnot XYZ, a to následovně [12]:

$$X = x_w X_W, \quad (1.18a)$$

$$Y = y_w Y_W, \quad (1.18b)$$

$$Z = z_w Z_W; \quad (1.18c)$$

$$x_w = \begin{cases} f_x^3, & \text{pokud } f_x^3 > 0,008856 \\ (f_x - 16/116)/7,787 & \text{jinak} \end{cases}, \quad (1.19a)$$

$$y_w = \begin{cases} f_y^3, & \text{pokud } f_y^3 > 0,008856 \\ (f_y - 16/116)/7,787 & \text{jinak} \end{cases}, \quad (1.19b)$$

$$z_w = \begin{cases} f_z^3, & \text{pokud } f_z^3 > 0,008856 \\ (f_z - 16/116)/7,787 & \text{jinak} \end{cases}; \quad (1.19c)$$

$$f_x = a^*/500 + f_y, \quad (1.20a)$$

$$f_z = f_y - b^*/200, \quad (1.20b)$$

$$f_y = (L^* + 16)/116; \quad (1.20c)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = [\mathbf{M}]^{-1} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}. \quad (1.21)$$

## 1.3 Šedotónový obraz v počítačové grafice

V dosud zmíněném textu se především hovořilo o barvách chromatických, tj. o barvách pestrých, jež mají určitý odstín a sytost. Avšak v mnoha případech se v počítači pracuje také s digitálními obrazy složenými z barev achromatických. Jsou to neutrální barvy, které se liší jen svou světlostí. K achromatickým barvám patří všechny odstíny šedé a také černá a bílá barva. Achromatický obraz – přesněji šedotónový – neobsahuje tedy informaci o barevnosti, ale pouze o jas. Pro označení achromatických barev v počítačové grafice se používá pojem „*stupně šedi*“ (angl. gray scale), šedotónové obrázky jsou pak často označovány jako obrázky ve *stupních šedi*.

Lidský vizuální systém vnímá jas barvy jako samostatný parametr, tedy nezávisle na informaci o její chromatičnosti. Proto obraz obsahující pouze jasovou složku – tedy šedotónový – se nám nezdá jako obraz neúplný nebo poškozený. Výhodou šedotónových obrazů je to, že k reprezentaci jasu v počítači postačuje pouze jedno číslo. Pro číselné vyjádření jasu jednotlivých pixelů v šedotónovém obraze se používají hodnoty v rozmezí od 0 do 255, kde 0 slouží jako kód pro nejtmaší odstín – černá barva – a maximální hodnota 255 udává nejsvětější odstín – bílá barva [13]. Stupně šedi mohou být také vyjádřeny čísly v rozsahu 0–1 nebo v procentech 0–100 % (minimální hodnota se rovná černé, maximální – bílé).

Pro převod barevného RGB obrazu na stupně šedi se používá vzorec (1.22), podle kterého se z jednotlivých barevných složek vypočítává jas (intenzita) výsledného obrazového bodu:

$$I = 0,299R + 0,587G + 0,114B. \quad (1.22)$$

Jednotlivé poměry složek jsou dané citlivostí lidského oka na každou ze základních barev RGB. Alternativou převodu RGB obrazu na stupně šedi je prostý průměr hodnot barevných složek:

$$I = \frac{R + G + B}{3}. \quad (1.23)$$

Stupně šedi jsou používány v oblasti digitálního zpracování a analýzy obrazu docela často, např. pro hodnocení kvality předání barevného tónu u digitálních fotografií<sup>5</sup>, při skenování či kopírování obrazu a také v tiskových procesech. Šedotónový obraz je pak vlastním výstupem mnoha zobrazovacích metod (např. rentgenové či MRI snímky, snímky z SEM mikroskopu atd.), a to především z důvodu, že přidáním údajů o barvě se výpočet obrazových dat obvykle podstatným způsobem

---

<sup>5</sup>V oblasti digitální fotografie se např. často můžeme setkat s tzv. neutrální šedou (angl. middle gray), která je zde obzvlášť důležitá, jelikož podle ní lze velice jednoduše srovnat všechny ostatní barvy [14].

komplikuje [13]. V případě, když informace reprezentovaná na výstupu šedotónovým obrazem stále není postačující, využívá se umělého barvení obrazu.

## 1.4 Nepravé barvy

*Nepravé barvy* někdy také *pseudobarvy* (angl. false colors, pseudo colors) jsou umělé barvy vytvořené pomocí barevné syntézy. Nepravé barvy neodpovídají přirozenému vjemu lidského oka a člověk není schopen je rozpoznat ve viditelném spektru, neboť tyto barvy ve skutečnosti ani neexistují. Nepravé barvy umožňují v obraze zaznamenat informaci, kterou není lidský vizuální systém schopen vnímat, čehož se s výhodou využívá zejména v oblastech medicíny, astronomie, geodézie, v průmyslu a v mnoha dalších příbuzných oborech. Zobrazení snímků v nepravých barvách je velmi užitečnou metodou, když je potřeba znázornit data pořízená v okem neviditelných oblastech elektromagnetického spektra – např. při zachycení infračerveného či ultrafialového záření vesmírnými teleskopy. Nepravé barvy jsou účinné také při zkoumání rentgenových či MRI snímků, při zpracování satelitních snímků, jelikož pomáhají zvýraznit skryté detaily obrazu, což vede k podstatnému zvýšení jeho diagnostické výtěžnosti.

### 1.4.1 Pseudobarvení

Zvláštním případem použití nepravých barev je *pseudobarvení*. Jedná se o způsob zpracování šedotónových obrazů, při kterém se odstíny šedi převádí na barevné odstíny. Technika pseudobarvení se uplatňuje v případech, když jsou získaná data na výstupu reprezentována černobílým snímkem – např. rentgenovým nebo termovizním – s tím se lze často setkat i ve výše zmíněných oborech. Zobrazení v nepravých barvách výrazně zlepšuje čitelnost šedotónových obrazů, neboť jeho využití je založeno na skutečnosti, že člověk dokáže rozeznat daleko více barevných odstínů než odstínů šedi [15].

Ve své nejzákladnější podobě lze pseudobarvení představit jako transformační funkci  $\mathbf{T}$ , pro kterou platí:

$$c(x, y) = \mathbf{T}(f(x, y)), \quad (1.24)$$

kde  $f(x, y)$  je původní obraz ve stupních šedi a  $c(x, y)$  je výsledný barevný vektor tvořený hodnotami barevných složek modelu RGB [16]. Transformační algoritmus se pak liší podle konkrétní zvolené metody.

Nejjednodušším příkladem, jak lze provést umělé barvení obrazu, je aplikace tzv. *barevné palety*, angl. color palette (též *barevná mapa*, angl. color map). Barevná paleta je obvykle reprezentována maticí (nebo polem) o třech sloupcích, z nichž je

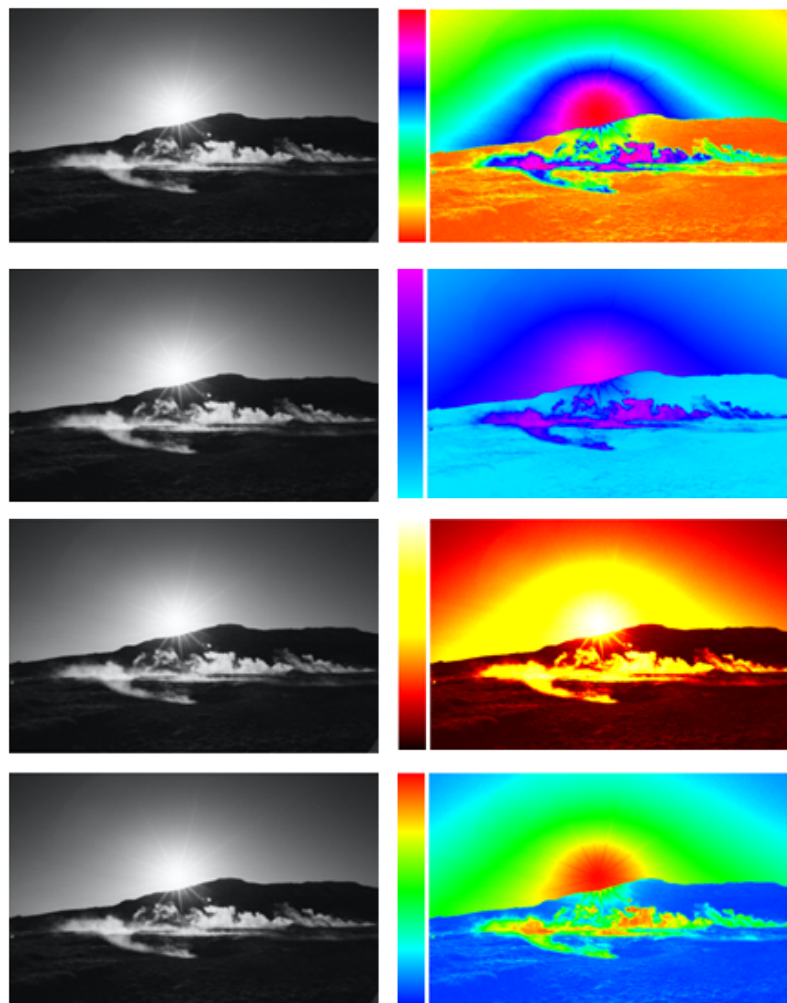
každý vyhrazen pro jednu z barevných složek RGB. Každý řádek v matici pak definuje konkrétní barvu z palety [17]. Principem metody barevné palety je vyhledání a volba řádku matice, a tím odpovídající barvy výstupního obrazového bodu, v závislosti na hodnotě jasu vstupního obrazového bodu v šedotónovém obraze. Jediným vstupem dané metody jsou tedy jasové hodnoty jednotlivých pixelů v originálním snímku. Metoda nezavádí další vstupní parametry a tím pádem se nemění obsah původních dat – při stejné hodnotě jasu bude stejná i výstupní hodnota [16]. Přiřazení barev dle zvolené barevné palety prakticky lze realizovat pomocí vyhledávací tabulky barev LUT<sup>6</sup> (nebo také CLUT - angl. color lookup table), která vyjadřuje závislost mezi pixely vstupního a výstupního obrazu. Principem sestavení této tabulky je použití indexů odkazujících na odpovídající barevnou položku palety (řádek v matici), která definuje barvu pixelu na výstupu. Jako indexy se v této tabulce používají hodnoty jasových úrovní vstupních pixelů šedotónového obrazu. Počet řádků tabulky je dán počtem diskretních jasových hodnot, kterých může nabývat pixel vstupního snímku, což se v případě šedotónového obrazu rovná 256.

LUT			
Jas vstupního pixelu	Výstupní hodnota barevné složky R	Výstupní hodnota barevné složky G	Výstupní hodnota barevné složky B
index	R(i)	G(i)	B(i)
0	R(0)	G(0)	B(0)
1	R(1)	G(1)	B(1)
2	R(2)	G(2)	B(2)
3	R(3)	G(3)	B(3)
...	...	...	...
...	...	...	...
...	...	...	...
249	R(249)	G(249)	B(249)
250	R(250)	G(250)	B(250)
251	R(251)	G(251)	B(251)
252	R(252)	G(252)	B(252)
253	R(253)	G(253)	B(253)
254	R(254)	G(254)	B(254)
255	R(255)	G(255)	B(255)

Obr. 1.9: Vyhledávací tabulka LUT.

<sup>6</sup>V praxi je tabulka LUT často představena maticí či datovým polem  $4 \times 256$ .

Obsah barevné palety se zpravidla volí podle jednotlivých účelů a potřeb a může být poměrně různorodý. Obr. 1.10 ukazuje v praxi některé z nejrozšířenějších barevných palet, jež se využívají pro demonstraci umělého barvení šedotónových obrazů v appletu „Převod šedotónového obrazu na barevný (nepravé barvy)“.



Obr. 1.10: Ukázka převodu šedotónového obrazu do nepravých barev pomocí různých barevných palet: HSV, cool, hot, Jet.



## 2 Praktická část

Daná kapitola se věnuje podrobnému popisu implementace jednotlivých appletů vytvořených v rámci této bakalářské práce, včetně ukázky základních funkcí zdrojových kódů a grafických uživatelských rozhraní appletů. Rovněž se v této části uvádí stručná charakteristika programovacích jazyků a vývojového prostředí, které byly použity při tvorbě appletů.

### 2.1 Software

Všechny applety mají fungovat jako jednoduchá webová stránka a vybízet uživatele k interakci. Požadované applety jsou tedy napsány v jazyce HTML s využitím kaskádových stylů CSS. Pro naprogramování ovládacích a dalších interaktivní prvků appletů je použit JavaScript. Právě s těmito technologiemi se v dnešní době můžeme na webu setkat nejčastěji.

Programy tvořené pomocí daných jazyků jsou relativně nenáročné a mohou být spouštěny na téměř každém zařízení podporujícím přístup na internet. Většina moderních webových prohlížečů nevyžaduje instalaci přídatných rozšíření či změnu základní konfigurace, takže vytvořené applety mohou být přístupné běžnému uživateli pro využití na webu.

#### 2.1.1 HTML a CSS

HTML je značkovací jazyk, který se používá pro tvorbu webových stránek. Popisuje především strukturu a obsah webu a to pomocí tzv. tagů, které označují jednotlivé elementy na webové stránce. Ve své nejnovější páté verzi HTML5 zahrnuje element `<canvas>` sloužící většinou k dynamickému vykreslování 2D grafiky pomocí JavaScriptu a samotného HTML5 skrze Canvas API [18]. Ve vytvořených appletech byl prvek `<canvas>` využit k načítání dat ze vstupních obrazů a k vykreslení obrazů na výstupu, a také pro vykreslování grafů.

Pro úpravu rozhraní webové stránky a nastavení způsobu zobrazení jejích jednotlivých komponentů se používají kaskádové styly CSS, které v podstatě stylizují celkový vzhled HTML stránky. V rámci appletů jsou však CSS použity spíše v minimální podobě pro zachování jednoduchosti.

#### 2.1.2 JavaScript

JavaScript je interpretovaný, multiplatformní, objektově orientovaný programovací jazyk, jehož autorem je Brendan Eich. Od jiných interpretovaných programovacích

jazyků užívaných na webu se JavaScript liší především tím, že nejčastěji běží na straně klienta, nikoli serveru, jako je tomu např. u jazyků PHP a ASP [19]. V současné době slouží většinou jako doplněk k HTML. Nejčastěji se používá zejména pro tvorbu scriptů řídících interaktivní a multimediální prvky na webové stránce. Pomocí JS se např. vytváří reakce na stisknutá tlačítka, upozornění při špatně zadaných datech do formulářů, dynamické stylování a animace [20].

JavaScript je relativně přátelský pro začátečníky díky své jednoduché syntaxi. Mezi dalšími výhodami jazyku lze také uvést jeho širokou použitelnost, pravidelnou aktualizaci a rychlost běhu díky tzv. just-in-time kompilaci.

### 2.1.3 Chart.js

Pro vykreslení veškerých grafů a nastavení jejich parametrů v appletu „Převod fyzikálního světla do barevných prostorů CIE XYZ a CIELAB“ se používá knihovna Chart.js (verze 3.1.0).

Je to malá, jednoduchá, ale poměrně flexibilní knihovna s otevřeným zdrojovým kódem napsaná v jazyce JavaScript [21]. Chart.js je jednou z nejpoblárnějších JS knihoven používaných pro tvorbu grafů [22]. Je dokonalým řešením pro malé nenáročné projekty, do nichž lze zařadit i webové aplikace vytvořené v rámci této bakalářské práce. Takže pro tvorbu grafů knihovna Chart.js se ukázala jako velice vhodná.

Celkem tato knihovna podporuje 8 typů grafů, lze přizpůsobit podle potřeb. Funkčnost grafů může být rozšířena pomocí různých doplňkových pluginů. Grafy je také možné pomocí Chart.js animovat. Při použití na webu jsou všechny typy grafů responzivní. Pro vykreslování grafů se používá již zmíněný element HTML5 `<canvas>`.

Dalšími výhodami knihovny Chart.js jsou dobře zpracovaná dokumentace, podpora přes známé programátorské fórum StackOverflow, podpora knihovny v prohlížečích IE9+ [21].

### 2.1.4 Editor zdrojového kódu VSCode

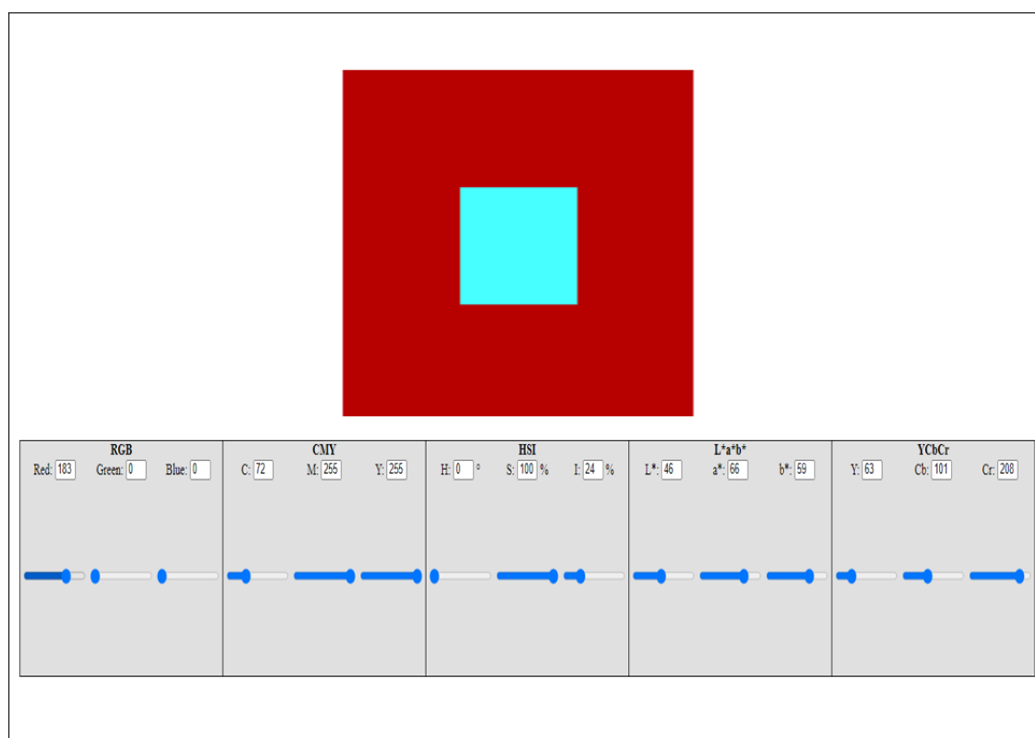
Jako vývojové prostředí pro tvorbu appletů byl použit editor zdrojového kódu Visual Studio Code od společnosti Microsoft, která jej poskytuje k volnému stažení na svých webových stránkách [23]. Daný program je velmi uživatelsky přívětivý: umožňuje jednoduché formátování psaného kódu, umí napovědět a automaticky doplnit potřebné části kódu, navíc obsahuje podporu zvýraznění syntaxe zobrazením jejích různých částí v různé barvě. VSCode je navržen a optimalizován primárně pro vývoj a ladění webových aplikací. Velkou výhodou programu je možnost spustit zdrojový kód v režimu „Live Server“, který umí zobrazit obsah vytvořeného appletu

ve webovém prohlížeči pro okamžité otestování kompatibility a ověření funkčnosti appletu.

## 2.2 Applet „Míchání barev v různých modelech“

Daný applet má za cíl názorně demonstrovat princip míchání barev v barevných modelech a prostorech popsaných v teoretické části této bakalářské práce.

Základními ovládacími prvky appletu jsou slidery, s jejichž pomocí lze nastavovat jednotlivé složky barevných modelů RGB, CMY, HSI,  $YCbCr$  a prostoru  $L^*a^*b^*$  a tím měnit výslednou barvu. Slidery jsou pak umístěny do jednotlivých sekcí vyhrazených pro každý z barevných modelů. Hodnoty barevných složek lze také zadat ručně pomocí klávesnice. Aktuální nastavované hodnoty se ukazují nad každým ze sliderů. Všechny barevné modely jsou navzájem propojeny, takže při změně jednotlivých barevných složek v jednom modelu se upravují parametry i v ostatních modelech. Tato vlastnost poskytuje uživateli možnost porovnat hodnoty jednotlivých parametrů stejné barvy v různých modelech. Míchaná barva je zobrazována v horní části okna uvnitř ohraničené oblasti. Zároveň se v této oblasti zobrazuje i barva doplňková k aktuální nastavené barvě.



Obr. 2.1: Ukázka appletu „Míchání barev v různých modelech“.

Vstupní hodnoty ze sliderů se načtou v modelu RGB. Pak se tyto hodnoty převedou do ostatních modelů. Přepočítání mezi modely je ve zdrojovém kódu prováděno pomocí metody **recalc()**, která volá jednotlivé funkce, jež definují algoritmus převodu barvy z modelu RGB do dalších zmíněných barevných modelů a zpět. Při naprogramování funkcí pro přepočítání jednotlivých složek RGB modelu na základní parametry modelů CMY, HSI,  $Y_C B_C R$  a prostoru  $L^*a^*b^*$  se vycházelo ze vzorců uvedených v sekci „Převod barev mezi barevnými modely a prostory“.

---

Výpis 2.1: Ukázka kódu funkce pro převod z RGB na  $Y_C B_C R$ .

---

```
function RGBtoYCbCr(basic){
  var r = basic.r/255,
      g = basic.g/255,
      b = basic.b/255,
      y,cb,cr;

  y = 16 + 65.481*r + 128.553*g + 24.966*b;
  cb = 128 - 37.797*r - 74.203*g + 112*b;
  cr = 128 + 112*r - 93.786*g - 18.214*b;

  basic.y2 = Math.round(y);
  basic.cb = Math.round(cb);
  basic.cr = Math.round(cr);
}
```

---

Aktuální hodnoty barevných složek všech modelů (a tím pádem zobrazovaná barva na výstupu) se nastavují pomocí metody **set()**, která definuje vlastnost objektu (tedy v daném případě hodnotu barevné složky) pomocí funkce, která má být volána při pokusu nastavení této vlastnosti.

---

Výpis 2.2: Ukázka kódu pro nastavení barevných složek modelu RGB.

---

```
setR : function(val){ // červená složka RGB
  if (isNaN(val)) return;
  this.r = Number(val);
  recalc(true);
  recalc({hsi: true, ycbcr: true, lab: true});}
```

---

## 2.3 Applet „Převod šedotónového obrazu na barevný (nepravé barvy)“

Daný applet má za cíl demonstrovat princip umělého barvení šedotónového obrazu za pomoci několika druhů barvicích palet.

Applet poskytuje možnost výběru ze čtyř barevných palet: Jet, cool, hot a HSV (viz obr. 1.10). Lze také využít možnost barvení obrazu pomocí lineárního RGB gradientu, kdy uživatel může ovlivnit barevnou paletu výsledného obrazu nastavením hodnot jednotlivých kanálů RGB. Zaškrtnutím políčka „Black“ nebo „White“ lze měnit hodnoty koncových bodů výsledné palety (viz obr. 2.2).

Výpis 2.3: Ukázka kódu pro výpočet výsledné RGB palety v závislosti na barvě koncových bodů.

```
if(black){ // koncový bod = černá
data[i] = endR - endR * r; // červená složka
data[i + 1] = endG - endG * g; // zelená složka
data[i + 2] = endB - endB * b; // modrá složka
}
else{ // koncový bod = bílá
data[i] = (255 - endR) * r + endR; // červená složka
data[i + 1] = (255 - endG) * g + endG; // zelená složka
data[i + 2] = (255 - endB) * b + endB; // modrá složka
}
```



Obr. 2.2: Ukázka appletu pro převod šedotónových obrazů do nepravých barev.

K barvení se nabízí několik šedotónových obrazů, které si uživatel může vybrat z rozbalovacího seznamu v horní části okna. Rovněž jsou tady umístěné veškeré ovládací prvky.

Stisknutím tlačítka „Provést přebarvení“ se na vybraný šedotónový snímek aplikuje zvolená paleta a vedle původního obrázku se objeví výstupní obraz v nepravých barvách. Mezi vstupním a výstupním obrazem se pak nacházejí ukázky původní šedotónové a výsledné barevné palety.

Vstupní snímek se zpracovává pomocí HTML5 elementu `<canvas>`, konkrétně pak pomocí metody `getImageData`, která umožňuje načíst obrazová data na úrovni jednotlivých pixelů. Barvení je pak prováděno pixel po pixelu dle následujícího algoritmu.

---

Výpis 2.4: Ukázka přiřazení barev dle barevné palety Hot.

---

```
function toHotConvert(){
    let canvas = document.getElementById('canvas');
    let ctx = canvas.getContext('2d');
    let img = document.getElementById('img');
    ctx.drawImage(img, 0, 0, canvas.width, canvas.height);
    let imageData = ctx.getImageData(0, 0, canvas.width,
    canvas.height);
    let data = imageData.data;
    for(let i = 0; i < data.length; i+=4){
        let r = data[i] / 255; //hodnoty červené složky
        r = r <= 0.25 ? 255 * r * 4 : 255;
        data[i] = r;
        let g = data[i + 1] / 255; //hodnoty zelené složky
        if(g <= 0.25) g = 0;
        else if(g <= 0.5) g = 255 * (g - 0.25) * 4;
        else g = 255;
        data[i + 1] = g;
        let b = data[i + 2] / 255; //hodnoty modré složky
        b = b <= 0.75 ? 0 : 255 * (b - 0.75) * 4;
        data[i + 2] = b;
    }
}
```

---

Algoritmus postupně přiřazuje každému obrazovému bodu jednotlivé hodnoty RGB složek podle zvolené palety. Samotná barevná paleta je definována přímo uvnitř algoritmu. Na výstupu dostáváme obarvený obraz, jenž je vykreslen zase pomocí elementu `<canvas>`. Uvedený princip platí i pro ostatní barvicí palety.

Applet navíc nabízí možnost nastavit gama korekci obarveného obrazu. Gama korekce je funkcí sloužící k transformaci jasových hodnot obrazu. V praxi bývá např. používána v případě, když je potřeba upravit příliš přepálenou nebo naopak nedostatečně osvětlenou oblast či objekt v snímku [15]. Při změně parametru gamy se mohou v obrazu objevit nové detaily, které předtím nebyly patrné.

Hodnoty všech RGB složek ve výsledné paletě po aplikaci gama korekce se přepočítávají podle vztahu:

$$newColor = 255 \cdot ((oldColor/255)^{1/\gamma}), \quad (2.1)$$

kde  $\gamma$  je číslo zadané uživatelem v rozsahu 0,1–10,0. Ve zdrojovém kódu tento vztah vypadá následovně.

---

Výpis 2.5: Ukázka implementace gama korekce.

---

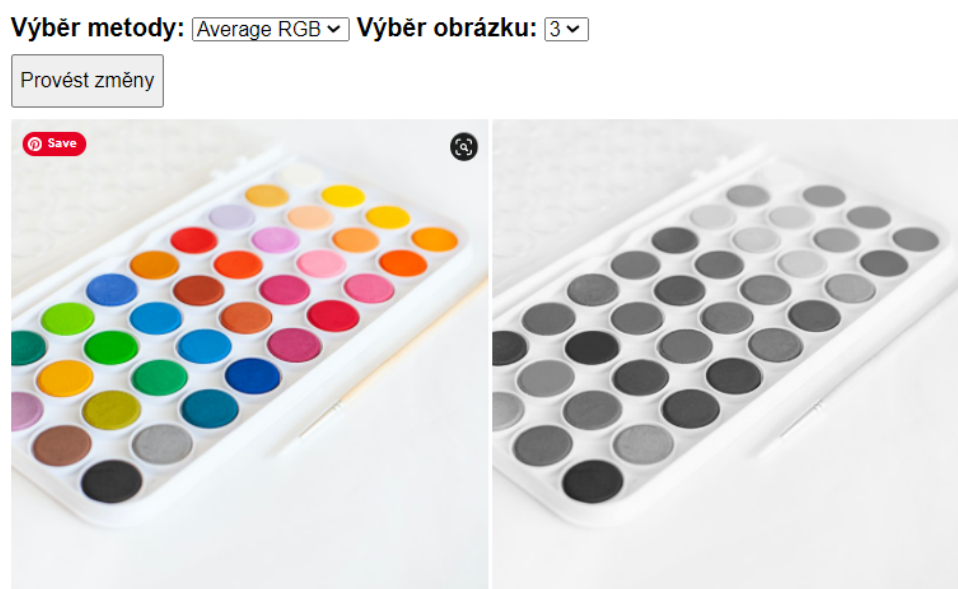
```
let r = data[i] / 255; // červená složka
data[i] = Math.pow(r, 1 / gamma) * 255;
let g = data[i + 1] / 255; // zelená složka
data[i + 1] = Math.pow(g, 1 / gamma) * 255;
let b = data[i + 2] / 255; modrá složka
data[i + 2] = Math.pow(b, 1 / gamma) * 255;
```

---

## 2.4 Applet „Převod barevného obrazu na šedotónový“

Daný applet má za cíl demonstrovat princip převodu barevného obrazu na obraz ve stupních šedi za pomoci různých metod.

K převodu se nabízí několik barevných obrazů, které si uživatel může vybrat z rozbalovacího seznamu v horní části okna. Dál si uživatel může zvolit metodu, podle níž se vstupní obraz převede na stupně šedi.



Obr. 2.3: Ukázka appletu pro převod barevného obrazu na šedotónový.

Po stisknutí tlačítka „Provést změny“ se vedle zvoleného barevného obrázku vykreslí výstupní šedotónový obrázek.

Stejně jako v předchozím appletu se vstupní obrazová data zde načítají a zpracovávají pixel po pixelu pomocí HTML5 elementu `<canvas>`. Přepočít barevných složek RGB obrazu se provádí pomocí funkce `converter()`, která pak pomocí konstrukce `switch()...case` na základě metody vybrané uživatelem provede příslušný algoritmus výpočtu hodnot pixelů výstupního obrazu.

### Metoda „Lumino“

Na základě dané metody vzniká šedotónový obraz, ve kterém je intenzita (jas) jednotlivých šedotónových pixelů vypočítána podle vztahu (1.22). Poměry RGB složek jsou v (1.22) dány citlivostí lidského oka na jednotlivé barevné složky, kde nejcitlivěji vnímá změnu zelené složky a nejméně vnímá změnu modré složky.



```
function lumino(){
    ...
    let data = imageData.data;
    for(let i = 0; i < data.length; i+=4){
        let r = data[i]; // red
        let g = data[i + 1]; // green
        let b = data[i + 2]; // blue
        let intensity = 0.299*r + 0.587*g + 0.114*b;
        data[i]= data[i +1] = data[i + 2] = intensity;
    }ctx.putImageData(imageData, 0, 0); }
```

---

### Metoda „Average RGB“

Daná metoda převede obraz na odstíny šedi průměrováním hodnot jednotlivých RGB složek vstupního obrazu podle rovnice (1.23). Výsledná intenzita šedotónových pixelů se vypočítává podle následujícího algoritmu.

```
function drawBW(perc){
    ...
    let data = imageData.data;
    for(let i = 0; i < data.length; i+=4){
        var avg = (perc[0] * data[i] +
        perc[1] * data[i + 1] + perc[2] * data[i + 2]) / 3;
        data[i]      = avg;
        data[i + 1] = avg;
        data[i + 2] = avg;
    };
    ctx.putImageData(imageData, 0, 0); }
```

---

### Metoda „LAB“

Daná metoda používá pro nastavení intenzity pixelů v šedotónovém obraze jasovou složku L v barevném prostoru  $L^*a^*b^*$ . Hodnoty složky L se vypočítají na základě vstupních hodnot v modelu RGB pomocí funkce **convertPixeltoLAB(rgb)**. Pak

po nastavení nějaké hodnoty parametru L zase dojde k přepočtu na model RGB pomocí funkce **convertPixelToRGB(lab)**, a tím pádem dostáváme výsledný pixel.

Při výběru této metody se na obrazovce objeví slider, s jehož pomocí kterého lze měnit parametr L v rozsahu 0–100% a tím ovlivnit výsledný jas šedotónového obrazu.

### Metoda „% RGB“

Daná metoda poskytuje možnost ovlivnit výslednou intenzitu pixelů v šedotónovém obrazu pomocí nastavení váhy jednotlivých RGB kanálů. V případě označení této metody se nad vstupním obrazem objevují tři slidery – jeden pro každý barevný kanál. Pomocí sliderů tedy lze měnit podíl jednotlivých RGB složek v rozsahu 0 až 100%. Sečtením hodnot nastavených na každém z kanálů RGB dostaneme výslednou intenzitu pixelů ve stupních šedi. V kódu je daná metoda představená pomocí funkce **percent(perc)**.

---

Výpis 2.8: Ukázka kódu metody „% RGB“.

---

```
function percent (perc){
    ...
    let data = imageData.data;
    for(let i = 0; i < data.length; i+=4){
        var avg = (perc[0] * data[i] +
            perc[1] * data[i + 1] + perc[2] * data[i + 2]);
        data[i] = avg;
        data[i + 1] = avg;
        data[i + 2] = avg;
    };
    ctx.putImageData(imageData, 0, 0); }
```

---

### Metoda „Lightness“

Podle této metody se výsledná intenzita šedotónového pixelu vypočítá jako průměr maximální a minimální hodnoty RGB složek ve vstupním pixelu. Příslušná funkce v kódu vypadá následovně:

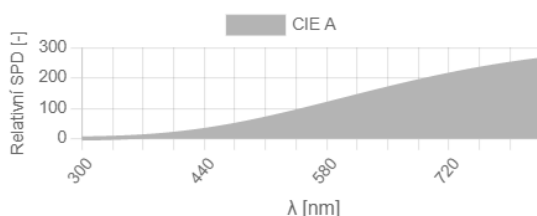
```
function lightness(){
    ...
    let data = imageData.data;
    for(let i = 0; i < data.length; i+=4){
        let r = data[i]; // red
        let g = data[i + 1]; // green
        let b = data[i + 2]; // blue
        let min = 0;
        let max = 0;
        ...
        let aver = (min + max) / 2;
        data[i] = data[i + 1] = data[i + 2] = aver;
    }
    ctx.putImageData(imageData, 0, 0);}

```

## 2.5 Applet „Převod fyzikálního světla do barevných prostorů CIE XYZ a $L^*a^*b^*$ “

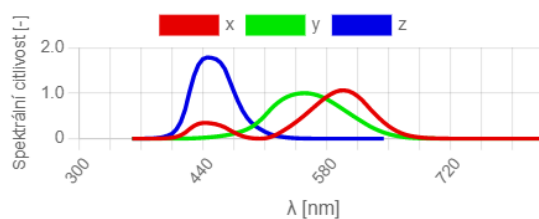
Daný applet má za cíl názorně ukázat, jakým způsobem se fyzikální světlo převádí na číselné hodnoty tristimulu CIE XYZ, z něhož se pak odvodí hodnoty barevných složek v prostoru  $L^*a^*b^*$  modelujícím lidské vnímání barev. Pomocí daného appletu má uživatel možnost získat přehled o komponentech, které ovlivňují výpočet XYZ.

Základ appletu tvoří grafy: graf funkce standardního osvětlení, grafy barvových funkcí standardního pozorovatele a graf zobrazující spektrum odrazivosti perfektního difuzéru<sup>1</sup>.

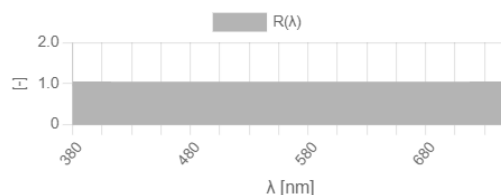


Obr. 2.4: Graf funkce standardního osvětlení CIE A.

<sup>1</sup>Ideální bílý povrch, který odráží 100% dopadajícího světla. Funkce odrazivosti perfektního difuzéru se rovná 1 při libovolné délce dopadajícího světla.

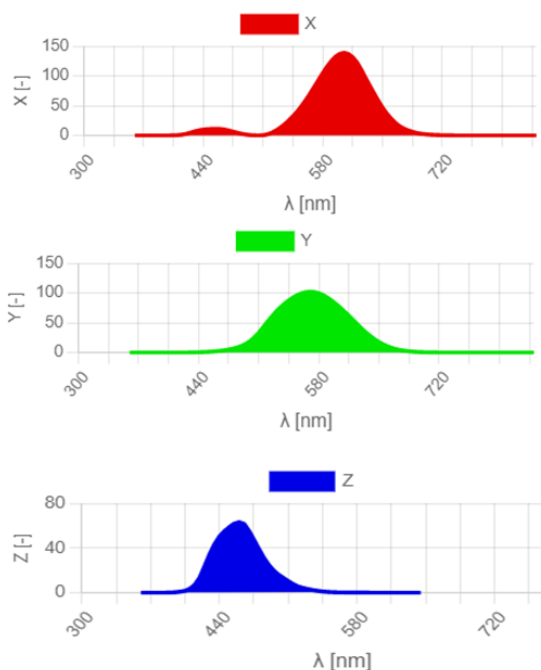


Obr. 2.5: Graf funkce standardního pozorovatele CIE.



Obr. 2.6: Spektrum odrazivosti perfektního difuzéru.

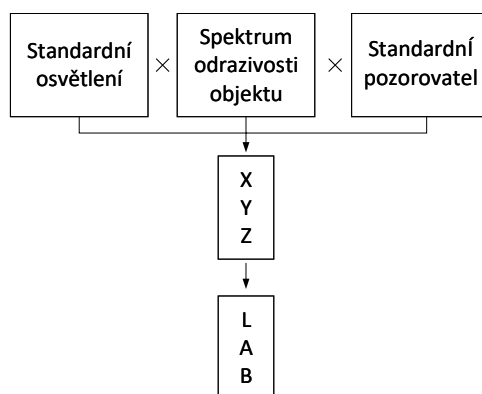
Data pro sestavení veškerých grafů jsou uložena do samostatného skriptu `data.js`, který je pak součástí zdrojového kódu appletu. Na základě těchto vstupních údajů se vypočítají hodnoty jednotlivých barevných složek XYZ, které jsou v appletu reprezentovány jak graficky (obr. 2.7) tak i trojicí čísel vyjadřujících barvu v prostoru CIE XYZ. Veškeré grafy se vykreslují pomocí metody **`drawGraf(canvas, type)`**.



Obr. 2.7: Grafické vyjádření tristimulu XYZ v appletu.

Nastavením určitého standardního osvětlovacího zdroje CIE dojde ke změně hodnot jednotlivých složek XYZ. Vybrat konkrétní zdroj osvětlení lze z rozbalovacího seznamu v horní části stránky. Pro výběr jsou nachystány zdroje A, C, B, D50, D65, F2, F7 a F12 (podrobněji lze o těchto osvětlovacích zdrojích přečíst v teoretické části, konkrétně pak v sekci „Prostor CIE XYZ“). Rovněž zde může uživatel zvolit i standardního pozorovatele CIE a tím zase ovlivnit výsledné hodnoty komponent XYZ a obsah jejich grafů.

Datová pole obsahující hodnoty jednotlivých složek XYZ potřebných pro vytvoření grafů na obr. 2.7 se generují pomocí funkce **getCustomData(type)** v závislosti na vstupních hodnotách funkcí standardního osvětlovacího zdroje a standardního pozorovatele. Princip výpočtu obsahu polí (tedy výpočet hodnot XYZ) by pak šlo zjednodušeně znázornit pomocí následujícího schématu.



Obr. 2.8: Princip odvození hodnot základního tristimulu XYZ v prostoru CIE XYZ.

Trojice čísel reprezentující složky XYZ se vypočítá jako součet všech hodnot pro každou z barevných složek a po stisknutí tlačítka „Vypočítat XYZ“ se objeví v první tabulce umístěné pod grafy. V kódu tento výpočet proběhne v rámci funkce **generateOutputs()**. Rovněž se v této tabulce zobrazí hodnoty v prostoru  $L^*a^*b^*$  a navíc i v prostoru  $xyY$ . Jejich výpočet se provede podle rovnic (1.17), resp. (1.2a)–(1.2c). Uživatel si může všimnout, že hodnota jasové složky  $L$  v prostoru  $L^*a^*b^*$  stejně jako hodnota jasové složky  $Y$  v prostorech XYZ a  $xyY$  se vždy rovná 100. Důvodem je, že se ve všech případech výpočty provádí pro dokonale bílý objekt, který odráží 100% dopadajícího světla. Tím pádem na výstupu dostáváme ve všech zmíněných prostorech číselné vyjádření čisté bílé barvy, kterou vnímá člověk při různých zdrojích světla. Podobně lze objasnit nulové hodnoty barvonosných parametrů  $a^*$  a  $b^*$  v prostoru  $L^*a^*b^*$ .

Původně se plánovalo přidat do appletu možnost vybrat na vstupu objekty s rozličnými spektry odrazivosti. Bohužel se mi v rámci této práce nepodařilo získat potřebná spektrální data pro jakékoli skutečné materiály, jelikož spektrální odrazivost konkrétních vzorků má být změřena spektrofotometrem. Applet však nabízí možnost zadat vstupní hodnoty XYZ a zjistit odpovídající hodnoty základních barevných parametrů v prostorech  $L^*a^*b^*$  a  $xyY$ . K jejich výpočtu se používá script `calculator.js`, jenž je součástí zdrojového kódu appletu.

# Závěr

Daná bakalářská práce se zabývala vývojem aplikací pro podporu výuky teorie barev a zpracování barevných obrazů. Jedná se o applety napsané v jazyce JavaScript a HTML, které je možné spouštět jako webovou stránku v prohlížeči. V rámci zadání byla popsána příslušná teorie týkající se reprezentace barev a barevných obrazů v počítači a následně využita pro vytvoření požadovaných appletů.

Teoretická část se věnuje objasnění základních pojmů. Je zde popsán pojem obrazu a jeho reprezentace v počítačové grafice. Definuje se také pojetí barvy a její charakteristické vlastnosti. Rovněž se vysvětluje jakým způsobem vnímá barvy člověk. Dál se v této části zavádí pojmy barevný model a barevný prostor. Poté následuje popis různých barevných modelů a prostoru, mezi něž patří: RGB, CMY(K), HSI,  $Y C_B C_R$ , CIE XYZ, xyY, CIE  $L^*a^*b^*$ . Uvádí se také matematické vztahy pro převod barev mezi jednotlivými modely. Poslední část teoretického úvodu je zaměřená na šedotónový obraz a nepravé barvy.

V praktické části jsou podrobně popsány jednotlivé applety vytvořené v rámci této práce včetně ukázky navrženého uživatelského rozhraní a vysvětlení funkce podstatných částí zdrojového kódu. Také je zde krátce zmíněn software, jenž byl využit při vývoji appletů.

První applet je věnován barevným modelům zmíněným v teoretické části. Pomocí tohoto appletu se demonstruje princip míchání barev v jednotlivých modelech či prostorech a také se ukazuje to, jak se navzájem převádí hodnoty jejich základních barevných složek. Druhý applet ilustruje převod barevného obrazu na stupně šedi pomocí několika metod: průměrováním složek RGB, pomocí jasového parametru v prostoru  $L^*a^*b^*$ , mícháním jednotlivých RGB kanálů, průměrováním maximální a minimální hodnoty RGB a nastavením poměrů RGB složek dle citlivostí lidského oka. Třetí applet slouží pro demonstraci principu umělého barvení šedotónového obrazu, tj. převod do nepravých barev při použití barevné palety. Poslední čtvrtý applet znázorňuje princip odvození číselných hodnot reprezentujících měřené fyzikální barvy v prostorech CIE XYZ a CIE  $L^*a^*b^*$ , který modeluje lidské vnímání barev.

Lze říci, že v rámci dané bakalářské práce bylo dosaženo všech stanovených cílů. Při vytvoření appletů byl důraz kladen především na jejich funkcionalitu. Představené applety jsou plně funkční, mají dostatečnou úroveň interaktivity a mohou sloužit k zamýšleným účelům – tedy jako podpora pro výuku teorie barev a počítačového zpracování barevných obrazů. Podnětem k následnému modifikování appletů může být rozpracování vzhledu grafického uživatelského rozhraní.

# Literatura

- [1] ŽÁRA, Jiří, Bedřich BENEŠ, Jiří SOCHOR a Petr FELKEL. *Moderní počítačová grafika*. 2. vyd. Praha: Computer Press, 2005, 609 s. ISBN 80-251-0454-0.
- [2] GONZALEZ, Rafael C. a Richard E. WOODS. *Digital Image Processing*. 3. vyd. Upper Saddle River: Pearson Education, 2008, 954 s. ISBN 978-0-13-168728-8.
- [3] Sensitivity of the human eye to visible light. In: *Light Colour Vision* [online]. c2015–2021 [cit. 2021-2-1]. Dostupné z: <<https://lightcolourvision.org/resource-library/sensitivity-eye-colour-spectrum/>>
- [4] Barevný model. In: *Wikipedia: The Free Encyclopedia* [online]. San Francisco: Wikimedia Foundation, poslední aktualizace 1. 3. 2021 [cit. 2021-3-8]. Dostupné z: <[https://cs.wikipedia.org/wiki/Barevn%C3%BD\\_model](https://cs.wikipedia.org/wiki/Barevn%C3%BD_model)>
- [5] HSI Color Conversion - Imaging toolkit feature: The HSI color space. In: *Black Ice* [online]. Boca Raton: Black Ice Software, c1989–2021 [cit. 2020-10-26]. Dostupné z: <<https://www.blackice.com/colorspaceHSI.htm>>
- [6] DEBOER, Clint. HDMI Enhanced Black Levels, xvYCC and RGB. In: *Aaudioholics* [online]. Tampa: Aaudioholics, c1998–2021, April 16, 2008 [cit. 2020-10-26]. Dostupné z: <<https://www.audioholics.com/home-theater-calibration/hdmi-black-levels-xvyc-rgb>>
- [7] A Guide to Understanding Color. In: *X-Rite* [online]. Grand Rapids: X-Rite, c2016 [cit. 2021-4-14]. Dostupné z: <<https://www.xrite.com/learning-color-education/whitepapers/a-guide-to-understanding-color>>
- [8] KOHOUT, Václav. Historie a elementární základy teorie barev III. *Školská fyzika* [online]. Plzeň: Oddělení fyziky, KMT FPE ZČU, 29. 7. 2013, X(2) [cit. 2021-3-3]. ISSN 2336-2774. Dostupné z: <<https://sf.zcu.cz/cs/2013/2/8-historie-a-elementarni-zaklady-teorie-barev-iii>>
- [9] WHETZEL, Norm. XYZ - CIE Tristimulus Values. *Applications Note: Insight on Color* [online]. Reston: HunterLab, c2008, 8(1) [cit. 2021-5-12]. Dostupné z: <[https://support.hunterlab.com/hc/en-us/article\\_attachments/201360059/an04\\_96a.pdf](https://support.hunterlab.com/hc/en-us/article_attachments/201360059/an04_96a.pdf)>
- [10] Illustration of the CIE 1931 color space. In: *Wikipedia: The Free Encyclopedia* [online]. San Francisco: Wikimedia Foundation, 13 July 2007 [cit. 2020-11-8]. Dostupné z: <<https://commons.wikimedia.org/wiki/File:CIExy1931.svg>>



- [11] PIHAN, Roman. BAREVNÝ MODEL L\*A\*B. In: *Wayback Machine* [online]. San Francisco: Internet Archive, c2011 [cit. 2020-11-2]. Dostupné z: <[https://web.archive.org/web/20170720105800/http://fotoroman.cz/glossary/3\\_lab.htm](https://web.archive.org/web/20170720105800/http://fotoroman.cz/glossary/3_lab.htm)>
- [12] LINDBLOOM, Bruce. Math. *Bruce Lindbloom* [online]. c2001–2021, poslední revize 15 Jul 2013 [cit. 2021-5-30]. Dostupné z: <<http://www.brucelindbloom.com/>>
- [13] ŠRÁMEK, Jaromír, Ondřej RÁČEK, Martin SEDLÁŘ a Vojtěch MORNSTEIN. *Získávání a analýza obrazové informace* [online]. Brno: Masarykova univerzita, Lékařská fakulta, Biofyzikální ústav, 2011, 111 s. [cit. 2021-5-2]. Dostupné z: <<https://www.med.muni.cz/biofyz/Image/ucebnice.pdf>>
- [14] ELIÁŠ, Majo. Šedá tabulka k vyrovnaní bílé. In: *Milujeme fotografii* [online]. Brno: ZONER software, c1996–2018, 1. června 2015 [cit. 2021-3-3]. Dostupné z: <<https://www.milujemefotografii.cz/seda-tabulka-k-vyrovnani-bile>>
- [15] BOLEČEK, Libor. Program pro zobrazení černobílých snímků v nepravých barvách: Pseudo-colour Paging of the Monochromatic Picture. *Elektrorevue* [online]. Brno: International Society for Science and Engineering, 20.12.2010, 12(6) [cit. 2021-4-24]. ISSN 1213-1539. Dostupné z: <<http://www.elektrorevue.cz/cz/clanky/zpracovani-signalu/0/program-pro-zobrazeni-cernobilych-snimku-v-nepravych-barvach/>>
- [16] WELSH, Tomihisa, Michael ASHIKHMIN a Klaus MUELLER. Transferring color to greyscale images. *ACM Transactions on Graphics* [online]. New York City: The Association for Computing Machinery, 2002, 21(3), [cit. 2021-3-24]. ISSN 1557-7368. Dostupné z: <<https://doi.org/10.1145/566654.566576>>
- [17] Colormap. In: *MathWorks* [online]. Natick: MathWorks, c1994–2021 [cit. 2021-4-29]. Dostupné z: <<https://www.mathworks.com/help/matlab/ref/colormap.html#buq1hym>>
- [18] Canvas API. In: *MDN Web Docs* [online]. San Francisco: Mozilla, c2005–2021, Feb 19, 2021 [cit. 2021-3-3]. Dostupné z: <[https://developer.mozilla.org/en-US/docs/Web/API/Canvas\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API)>
- [19] JavaScript. In: *Wikipedia: The Free Encyclopedia* [online]. San Francisco: Wikimedia Foundation, 4.2.2021 [cit. 2021-3-24]. Dostupné z: <<https://cs.wikipedia.org/wiki/JavaScript>>

- [20] JILEČEK, Jan. 5 programovacích jazyků pro začátečníky. In: *Create IT* [online]. Praha: Cleverlance, c2021, 18.12.2019 [cit. 2021-1-2]. Dostupné z: <<https://www.create-it.cz/Blog/Stranky/5-jazyku-pro-2020.aspx>>
- [21] BOROVIKOV, Ruslan. Top 10 JavaScript Charting Libraries for Every Data Visualization Need. In: *Hacker Noon* [online]. Artmap, 2016, April 25th 2021 [cit. 2021-4-30]. Dostupné z: <<https://hackernoon.com/10-javascript-charting-libraries-data-visualization-b77523d23372>>
- [22] SHARMA, Nishith. The 14 best data visualization tools. In: *TNW* [online]. Amsterdam: Financial Times, c2006–2021, April 21, 2015 [cit. 2021-5-13]. Dostupné z: <<https://thenextweb.com/news/the-14-best-data-visualization-tools>>
- [23] Visual Studio Code [online]. Redmond: Microsoft, c2021 [cit. 2021-5-29]. Dostupné z: <<https://code.visualstudio.com/>>

# Seznam symbolů a zkratek

<b>ASP</b>	webový aplikační framework a programovací jazyk – Active Server Pages
<b>CIE</b>	Mezinárodní komise pro osvětlování – Commission Internationale de l'éclairage
<b>(C)LUT</b>	vyhledávací tabulka barev – Color Lookup Table
<b>CMY(K)</b>	azurová, fialová, žlutá, černá – Cyan, Magenta, Yellow, Black
<b>CSS</b>	kaskádové styly – Cascading Style Sheets
<b>HSI</b>	odstín, sytost, intenzita – Hue, Saturation, Intensity
<b>HTML</b>	značkovací jazyk – Hypertext Markup Language
<b>ITU-R</b>	Radiokomunikační sektor Mezinárodní telekomunikační unie – Radiocommunication Sector of the International Telecommunication Union
<b>JPEG</b>	metoda ztrátové komprese používané pro ukládání počítačových obrázků – Joint Photographic Experts Group
<b>[K]</b>	jednotka termodynamické teploty – Kelvin
<b><math>L^*a^*b^*</math></b>	$L^*$ – luminance, $a^*$ – osa zelená–červená, $b^*$ – osa modrá–žlutá
<b>MPEG</b>	standard používaný pro kódování audiovizuálních informací – Moving Picture Experts Group
<b>PHP</b>	skriptovací programovací jazyk – PHP: Hypertext Preprocessor
<b>RGB</b>	červená, zelená, modrá – Red, Green, Blue
<b>SPD</b>	spektrální rozložení energie osvětlení – Spectral Power Distribution
<b>sRGB</b>	barevný prostor – Standard RGB
<b>xyY</b>	$x$ a $y$ – souřadnice chromatičnosti, $Y$ – jas
<b>XYZ</b>	základní tristimulus CIE: $Y$ – jas, $X$ a $Z$ – specifikace barvy
<b><math>YC_B C_R</math></b>	$Y$ – jas, $C_B$ a $C_R$ – modrý a červený chrominanční komponent

## A Obsah přílohy

Příložený ZIP soubor obsahuje veškeré soubory potřebné ke spuštění jednotlivých appletů. Jedná se o soubory s příponou `.html`, `.js` a `.css`, které jsou součástí každého z vytvořených appletů. Applety „Převod šedotónového obrazu na barevný“ a „Převod barevného obrazu na šedotónový“ je nutné nejprve umístit na web, lokálně je nelze spustit z důvodu přístupu k obrázkům. Ostatní applety je také doporučeno nejdříve umístit na web pro správné zobrazení v prohlížeči.

Applety byly testovány na PC s operačním systémem Windows 10 na webových prohlížečích Google Chrome verze 91.0.4472.77 a Mozilla Firefox verze 88.0.1 (verze jsou aktuální k datu 31. 5. 2021).

```
/ ..... kořenový adresář přiloženého ZIP souboru
├── MichaniBarev ..... applet „Míchání barev v různých modelech“
│   ├── michani_barev.html ..... samotný applet
│   ├── js
│   │   └── script.js ..... řídicí script appletu
│   └── css
│       └── style.css ..... CSS styly appletů
├── NepraveBarvy ..... applet „Převod šedotónového obrazu na barevný“
│   ├── false_colors.html ..... samotný applet
│   ├── images ..... soubor s obrázky
│   ├── js
│   │   └── script.js ..... řídicí script appletu
│   └── css
│       └── style.css ..... CSS styly appletů
├── PrevodNaSedoton ..... applet „Převod barevného obrazu na šedotónový“
│   ├── gray_scale.html ..... samotný applet
│   ├── images ..... soubor s obrázky
│   ├── script.js ..... řídicí script appletu
│   └── style.css ..... CSS styly appletů
└── SvetloXYZLab ..... applet „Převod ... do prostorů CIE XYZ a L*a*b*“
    ├── xyz_lab.html ..... samotný applet
    ├── js
    │   ├── calculator.js ..... řídicí script appletu
    │   ├── data.js ..... script obsahující vstupní údaje appletu
    │   └── script.js ..... řídicí script appletu
    └── css
        └── style.css ..... CSS styly appletů
```